

Microsoft Cloud Adoption Framework for Azure

Proven guidance and best practices that help you confidently adopt the cloud and achieve business outcomes.



OVERVIEW
[Explore scenarios](#)



DOWNLOAD
[Get tools and templates](#)



WHAT'S NEW
[Find out what's new](#)



OVERVIEW
[Learn about the Framework](#)

Cloud Adoption Framework guidance

Find guidance for each phase of your cloud adoption journey.



Get started

- [Get started](#)
- [Accelerate migration](#)
- [Deliver operational excellence](#)
- [Antipatterns to avoid](#)



Strategy

- [Motivations](#)
- [Business outcomes](#)
- [Financial considerations](#)
- [Technical considerations](#)



Plan

- [Rationalize your digital estate](#)
- [Organizational alignment](#)
- [Skills readiness plan](#)
- [DevOps cloud adoption plan](#)



Ready



Migrate



Innovate

Operating model alignment
Azure landing zone conceptual architecture
Azure landing zone design areas
Implementation options

Overview
Checklist
Product migration scenarios

Business value consensus
Build your first MVP
Measure for customer impact
Expand digital inventions



Secure

Overview
Plan for a Secure Cloud Adoption
Integrate Security Into Your Cloud Adoption Strategy
Prepare Your Secure Cloud Estate



Manage

Business commitments
Management baseline
Expand the baseline
Advance operations and design principles



Govern

Overview
Checklist

Assessments

Plan your cloud journey with assessments that help you measure the difference between your current state and cloud adoption goals. Assessment results provide curated technical guidance to help you reach your goals.

Cloud Adoption Strategy Evaluator

- Build and advance your cloud business case with the cloud adoption strategy assessment.

App and Data Modernization Readiness

- Take the first step in modernizing your workloads by taking this application and data modernization assessment.

Strategic Migration Assessment and Readiness

- Evaluate your cloud migration readiness by using the Strategic Migration and Readiness Assessment Tool (SMART).

Cloud Governance

- Assess your cloud governance approach and receive tailored recommendations for improvement.

Build your skills with Microsoft Learn training

Learn more about Azure and the Cloud Adoption Framework by completing these learning paths.



[Microsoft Cloud Adoption Framework for...](#)



[Azure Fundamentals](#)



[Learn the business value of Azure](#)

Find help

Microsoft has dedicated resources to help unblock your cloud adoption journey.

[FastTrack for Azure](#)

Get support from Microsoft engineers.

[Azure Migrate and Modernize](#)

Get support migrating workloads to the cloud.

[Partner-led Workshops](#)

Obtain best practices on Cloud Adoption Framework from certified partners.

[Find a partner](#)

Find a partner to help you get started.

Why the Microsoft Cloud Adoption Framework for Azure is right for your business. <https://aka.ms/adopt>

What is the Microsoft Cloud Adoption Framework for Azure?

Article • 03/27/2023

The Microsoft Cloud Adoption Framework for Azure is a full lifecycle framework that enables cloud architects, IT professionals, and business decision makers to achieve their cloud adoption goals. It provides best practices, documentation, and tools that help you create and implement business and technology strategies for the cloud.

Following best practices for the Cloud Adoption Framework allows your organization to better align business and technical strategies and ensure success. Watch the following video to learn more.

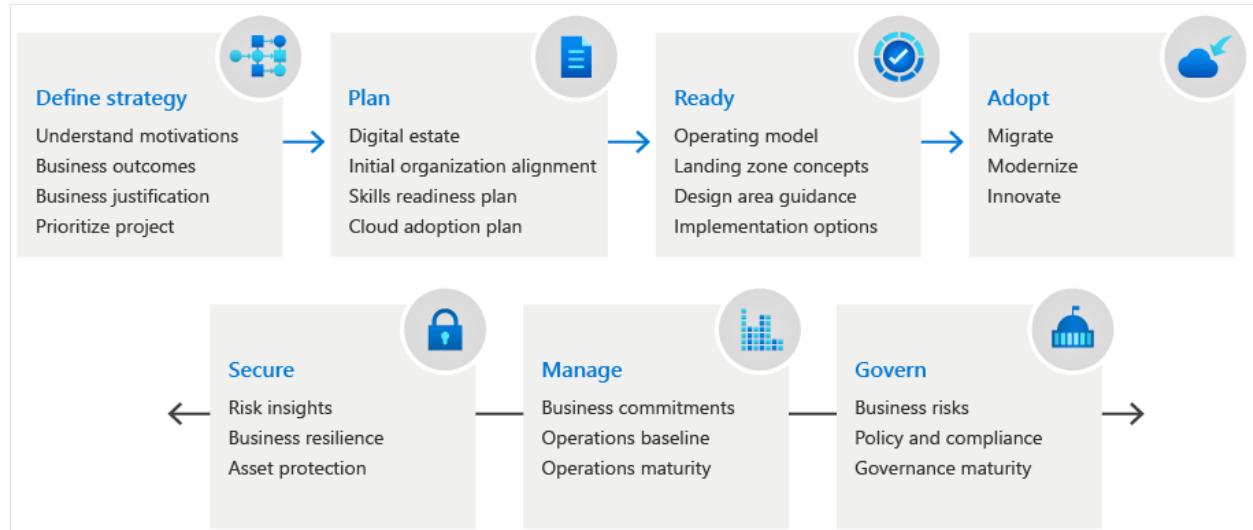
[https://www.microsoft.com/en-us/videoplayer/embed/RE4tyzr?postJslIMsg=true ↗](https://www.microsoft.com/en-us/videoplayer/embed/RE4tyzr?postJslIMsg=true)

The Cloud Adoption Framework brings together cloud adoption best practices from Microsoft employees, partners, and customers. The framework provides tools, guidance, and narratives. The tools it includes help you shape your technology, business, and people strategies to achieve the best business outcomes possible through your cloud adoption effort. Use the following table to review the guidance for each methodology.

Methodology	Description
 Strategy: Define business justification and expected adoption outcomes.	 Plan: Align actionable adoption plans to business outcomes.
 Ready: Prepare your cloud environment for planned changes.	 Migrate: Migrate and modernize existing workloads.
 Innovate: Develop new cloud-native or hybrid solutions.	 Secure: Improve security over time.
 Manage: Manage operations for cloud and hybrid solutions.	 Govern: Govern your environment and workloads.
 Organize: Align the teams and roles supporting your organization's cloud adoption efforts.	

Understand the lifecycle

Each methodology listed above is part of a broad cloud adoption lifecycle. The Cloud Adoption Framework supports you throughout each phase of your cloud adoption journey. The following diagram outlines how the framework uses methodologies as approaches to overcoming common blockers.



Intent

Cloud-based infrastructure fundamentally changes how your organization finds, uses, and secures technology resources. Traditionally, organizations assumed ownership of and responsibility for all aspects of their technology, from infrastructure to software. Moving to the cloud instead allows your organization to provision and consume resources only when needed. Although the cloud offers tremendous design choice flexibility, your organization needs a proven and consistent methodology for adopting cloud technologies to ensure success. The Microsoft Cloud Adoption Framework for Azure meets that need, helping guide your decisions throughout your cloud adoption journey.

Cloud adoption is a means to an end. Successful cloud adoption begins well before any cloud platform vendor is selected. It begins when business and IT decision makers realize that the cloud can accelerate a specific business transformation goal. The Cloud Adoption Framework helps decision makers align strategies for business, culture, and technical change to achieve desired business outcomes.

The Cloud Adoption Framework provides technical guidance for Microsoft Azure. Enterprise customers might still be trying to select a cloud vendor, or might have an intentional multicloud strategy. For these situations, the framework provides cloud-agnostic guidance for strategic decisions whenever possible.

Intended audiences

This guidance affects the business, technology, and culture of organizations. Affected roles include:

- Line-of-business leaders
- Business decision makers
- IT decision makers
- Finance
- Enterprise administrators
- IT operations
- IT security and compliance
- IT governance
- Workload development owners
- Workload operations owners
- Business subject matter experts

Each role uses unique vocabulary, and each has different goals and key performance indicators. A single set of content can never address all audiences effectively.

Enter the *cloud architect*. A cloud architect serves as a thought leader and facilitator, bringing these audiences together. This collection of guides is designed to drive decision-making and help cloud architects have the right conversations with the right audiences. Business transformation empowered by the cloud relies on the cloud architect role to help guide decisions throughout the organization and IT.

Each section of the Cloud Adoption Framework represents a different facet of the cloud architect role. These sections also create opportunities to share cloud architecture responsibilities across a team of cloud architects. For example, the governance section is designed for cloud architects who have a passion for mitigating technical risks. Some cloud providers refer to these specialists as *cloud custodians*. We prefer the term *cloud guardian*, or collectively, a *cloud governance team*.

Use the Microsoft Cloud Adoption Framework for Azure

If your organization is new to Azure, begin by ensuring you [understand and document foundational alignment decisions](#). When your enterprise's digital transformation involves the cloud, having an understanding of these fundamental concepts helps you during the cloud adoption process.

Next steps

[Get started](#)

What's new in the Microsoft Cloud Adoption Framework for Azure

Article • 12/03/2024

We build the Microsoft Cloud Adoption Framework collaboratively with our customers, partners, and internal Microsoft Teams. We release new and updated content for the framework as it becomes available. These new releases pose an opportunity for you to test, validate, and refine the Cloud Adoption Framework guidance along with us.

Partner with us in our ongoing effort to develop the Cloud Adoption Framework.

November 2024

New articles

- [Monitor a cloud environment](#): This month, we retired outdated content on cloud monitoring and introduced a new overview that provides comprehensive guidance in a simplified format. Explore the importance of monitoring, the key components of a monitoring strategy, and the tools and services you can use to monitor your cloud environment effectively.
- [Well-architected considerations for AI workloads on Azure infrastructure \(IaaS\)](#): Explore the importance of well-architected AI solutions and how to apply the Azure Well-Architected Framework to your AI workloads. Find guidance on reliability, security, cost optimization, operational excellence, and performance efficiency.

Secure methodology refresh

This month, we made significant updates to the Secure methodology. The Secure methodology provides guidance on how to secure your cloud environment and protect your data. The methodology includes the following articles:

- [Secure overview](#): Learn about the Secure methodology and how to apply it to secure your cloud environment. Explore the key components of the Secure methodology, including security principles, security controls, and security best practices.
- [Security teams, roles, and functions](#): Learn about the key security teams, roles, and functions that are essential for securing your cloud environment.

- [Integrate security into your cloud adoption strategy](#): Explore key considerations for integrating security into your cloud adoption strategy.
- [Plan for a secure cloud adoption](#): Learn about the key considerations for planning a secure cloud adoption and the tools and services you can use to plan for a secure cloud adoption.
- [Prepare your secure cloud estate](#): Find guidance on the key considerations for preparing your secure cloud estate and the tools and services you can use to prepare your cloud estate securely.
- [Perform your cloud adoption securely](#): Explore the importance of security in cloud adoption and the key considerations for securely adopting cloud services.
- [Securely govern your cloud estate](#): Find guidance on the key considerations for securely governing your cloud estate and the tools and services you can use to securely govern your cloud environment.
- [Manage your cloud estate with enhanced security](#): Explore the importance of managing your cloud estate with enhanced security and the key considerations for managing your cloud estate securely.

SAP and Power Platform

We introduced new articles that provide guidance on integrating SAP and Power Platform. Learn how to extend an SAP landing zone to support Power Platform, understand the architecture workflow, and explore the fundamentals of SAP and Power Platform integration.

- [SAP and Microsoft Power Platform architecture workflow](#): Find guidance on how to design, deploy, and manage an integrated SAP and Power Platform solution. Explore the key components of the architecture, including SAP systems, Azure services, and Power Platform components.
- [Extend an SAP landing zone to support Microsoft Power Platform](#): Learn how to extend an SAP landing zone to support Power Platform.
- [SAP and Power Platform fundamentals](#): Explore the fundamentals of integrating SAP and Power Platform. Learn about the benefits of integrating SAP and Power Platform, the key components of the integration, and the architecture considerations for a successful integration.

Updated articles

- [Azure API Management landing zone accelerator](#): Find new guidance on generative AI gateway scenarios and how to use them in your API Management landing zone as well as new architecture examples.

- [Introduction to Oracle on Azure adoption scenarios](#): Explore new guidance on multi-region design for Oracle workloads on Azure, enhancing availability, scalability, and disaster recovery.

October 2024

New articles

Azure VMware Solution and Global Reach

- [Establish Cross-Tenant Network Connectivity for Azure VMware Solution SDDCs](#): Learn how to establish cross-tenant network connectivity for Azure VMware Solution software-defined datacenters (SDDCs) using Azure Virtual WAN and network virtual appliances (NVAs). Explore connectivity options between SDDCs, Azure, and on-premises environments.
- [Use a Dual-Region Azure VMware Solution Design That Has Virtual WAN and Global Reach](#): Learn best practices for deploying Azure VMware Solution in two regions by focusing on secure connectivity, traffic flows, and high availability using Azure Virtual WAN and Global Reach.
- [Use a Dual-Region Azure VMware Solution Design That Doesn't Have Global Reach](#): Explore recommendations for network connectivity, traffic flows, high availability, and the configuration of various components like Virtual WAN hubs, Azure Firewalls, and routing intents.
- [Secure Virtual WAN for Azure VMware Solution in a Single Region or in Dual Regions](#): Learn how to design secure Virtual WAN topologies for Azure VMware Solution in both single and dual-region scenarios. Learn about routing intent for traffic inspection and explore design considerations for deployments with and without Azure ExpressRoute Global Reach.
- [Use a Single-Region Azure VMware Solution Design That Has Virtual WAN and Global Reach](#): Explore best practices for configuring a secure Virtual WAN with routing intent and Azure ExpressRoute Global Reach for a single-region Azure VMware Solution. Find guidance on network connectivity, traffic flows, and the configuration of security solutions in the Virtual WAN hub.
- [Use a Single-Region Azure VMware Solution Design That Doesn't Have Global Reach](#): See recommendations for configuring a single-region Azure VMware Solution with secure Virtual WAN and routing intent without using Azure ExpressRoute Global Reach. Find guidance on network connectivity, traffic flows, and security considerations for Azure VMware Solution private clouds, on-premises sites, and Azure-native resources.

New CAF Scenario: AI Adoption on Azure

- [Establish an AI Center of Excellence](#): Learn how to create and manage an AI Center of Excellence (AI CoE) to drive AI adoption within an organization. Find guidance on the importance of an AI CoE, defining its functions, building a cross-functional team, structuring operations, and ensuring ongoing monitoring and evolution of AI initiatives.
- [Recommendations for organizations governing AI workloads in Azure](#): Learn best practices and recommendations for integrating AI risk management into broader risk management strategies, assessing organizational AI risks, documenting and enforcing AI governance policies, and monitoring AI risks.
- [Recommendations for managing AI](#): Learn best practices for managing AI workloads in Azure, including AI operations, deployment, endpoint sharing, model management, cost management, data management, and business continuity. Explore the need for structured practices, continuous monitoring, and adherence to governance standards to ensure effective and reliable AI system management.
- [Recommendations for organizations planning AI adoption](#): See guidance on integrating AI into an organization, including assessing and acquiring AI skills, accessing AI resources, prioritizing AI use cases, creating AI proofs of concept, implementing responsible AI practices, and estimating delivery timelines.
- [Recommendations for organizations building AI workloads in Azure](#): Explore guidance on establishing reliability, governance, networking, and foundational infrastructure for AI workloads in Azure. Learn best practices for ensuring availability, managing costs, securing networks, and creating scalable environments.
- [Recommendations for organizations securing AI workloads in Azure](#): Review guidelines on assessing AI security risks, implementing security controls for AI resources and data, and maintaining these controls through continuous monitoring and updates. Learn about the importance of protecting the confidentiality, integrity, and availability of AI models and data to prevent breaches and ensure compliance.
- [Recommendations for organizations developing an AI adoption strategy](#): Learn the latest guidance on identifying AI use cases, defining technology and data strategies, and ensuring responsible AI practices to effectively adopt AI within an organization.

AI workloads on Azure infrastructure (IaaS)

- [Compute recommendations](#): Learn how to select virtual machines, images, and orchestration solutions to optimize AI workloads on Azure. See recommendations

for training and inferencing AI models, managing costs, and using containers for scalable AI solutions.

- **Implementation options:** See recommendations for deploying AI workloads using Azure CycleCloud and Slurm. This article covers cluster creation, dynamic management, and infrastructure control, offering guidelines and architecture for efficient AI operations on Azure IaaS.
- **Governance recommendations:** Explore guidelines for managing resources, controlling costs, ensuring security, and maintaining operational consistency for AI workloads on Azure.
- **Management recommendations:** Learn strategies for effectively managing AI workloads on Azure by emphasizing continuous monitoring, optimizing practices, and establishing robust backup and disaster recovery plans.
- **Networking recommendations:** Learn about how to network to optimize bandwidth, minimize latency, and implement high-performance networking for AI workloads on Azure. Explore strategies for resource placement, using proximity placement groups, and utilizing GPU-optimized VMs and InfiniBand for efficient data processing.
- **Security recommendations:** Find guidance on securing Azure services, networks, data, access, and operating systems for AI workloads. Learn how to prioritize encryption, network security, access control, and incident response preparation.
- **Storage recommendations:** Learn how to use different storage options like Azure Managed Lustre, Azure NetApp Files, and local NVMe/SSD-based storage for active data, transferring inactive data to Azure Blob Storage, implementing checkpointing for model training, automating data migration to lower-cost storage tiers, ensuring data consistency, and enabling data versioning for reproducibility.

AI workloads and Azure AI platform services (PaaS)

- **AI architecture guidance to build AI workloads on Azure:** This set of articles provides architecture guidance for building AI workloads on Azure using platform-as-a-service (PaaS) solutions, including references and guides for both generative and nongenerative AI architectures, as well as recommendations for AI resource selection, networking, governance, management, and security.
- **Governance recommendations:** Find recommendations and best practices for managing, including AI model governance, cost control, platform policies, security measures, operational management, regulatory compliance, and data governance.
- **Management recommendations:** Learn best practices for deployment, model monitoring, operations, data management, and business continuity to ensure effective and secure AI operations.
- **Networking recommendations:** Explore networking recommendations, including how to configure and secure virtual networks, manage connectivity, and

implement strategies to protect sensitive AI resources and ensure data integrity and privacy.

- [Resource selection recommendations](#): Find guidance on choosing the right Azure AI platform, compute resources, data sources, and processing tools for both generative and nongenerative AI applications.
- [Security recommendations](#): Learn security recommendations covering topics such as securing AI resources, models, access, and execution to protect against potential threats and maintain data integrity and compliance.

Updated articles

- We made updates to the Azure Landing Zone architecture diagram to reflect guidance for multi-region deployments:
 - [Azure landing zones for modern application platforms](#)
 - [Review your environment or Azure landing zone for an SAP enterprise-scale migration](#)
- We refreshed the cloud-scale analytics documentation. These updates include modifications to reflect the latest product naming and guidance and to improve clarity and readability. Explore the following articles to learn more:
 - [Development lifecycle](#)
 - [What is a data mesh?](#)
 - [Key considerations for Azure Data Lake Storage](#)
 - [Azure Data Lake Storage](#)
 - [Data lake zones and containers](#)
 - [Data quality](#)
 - [Metadata standards](#)
 - [Requirements for governing data](#)
 - [Cloud-scale analytics for regulated industries - Microsoft Cloud Adoption Framework for Azure](#)
 - [Introduction to cloud-scale analytics for regulated industries](#)
 - [Data privacy for cloud-scale analytics in Azure](#)
 - [Azure Well-Architected Framework for data workloads](#)
- [Enterprise-scale support for Azure Virtual Desktop](#): We made updates to the baseline architecture and guidance for Azure Virtual Desktop in the enterprise-scale landing zone.
- [Inventory and visibility in Azure](#): We made updates to the guidance for Azure Monitor Agent.

- [Operational compliance considerations](#): We reviewed the Azure Update Manager guidance and refreshed the architecture diagram.

September 2024

Updated articles

- [SAP on Azure landing zone accelerator](#): We added guidance on best practices for zone resiliency.

We reviewed and made updates to the following articles to reflect the latest product naming and guidance:

- [Data catalog](#)
- [Data lineage](#)
- [Data quality](#)
- [Manage master data](#)
- [Metadata standards](#)

We made updates to the following articles for clarity and accessibility:

- [Adopt responsible and trusted AI principles](#)
- [Business commitment in cloud management](#)
- [Migrate Oracle workloads to Azure](#)
- [Network connectivity for Azure Arc-enabled servers](#)
- [Security governance and compliance for Citrix on Azure](#)
- [Security guidelines for Oracle Database@Azure](#)

August 2024

Updated articles

- [Responsible and trusted AI adoption](#): Explore new guidance about how to develop AI responsibly and build safer systems with Azure AI Content Safety. Learn about the importance of responsible AI and the features of the Responsible AI dashboard for Azure Machine Learning.
- [Landing zone identity and access management](#): Find new guidance about built-in and custom RBAC roles, least-privilege access, and conditions.
- [Business commitment in cloud management](#): We updated the service-level agreement references to service-level objective references.

- [Security guidelines for Oracle Database@Azure](#): Explore new updates about network security group rules and Oracle Data Safe.
- [Business continuity and disaster recovery for an SAP migration](#): Learn about the benefits of using Azure Backup to back up databases that have SAP HANA System Replication enabled.

We made updates to the following articles for clarity:

- [Network considerations for Azure VMware Solution dual-region deployments](#)
- [What is a data product?](#)
- [Cloud adoption scenarios](#)
- [Oracle on Azure IaaS landing zone accelerator](#)
- [Security governance and compliance for Citrix on Azure](#)

July 2024

New articles

This month, we introduced new articles that have guidance for Red Hat Enterprise Linux (RHEL) on Azure. We also added new articles and made major updates to existing articles for Oracle on Azure IaaS and Oracle Database@Azure. Take a look at the new and updated content to see how you can apply these recommendations in your organization.

Red Hat Enterprise Linux on Azure

- [Azure RHEL landing zone accelerator](#): Learn how to use the RHEL landing zone accelerator to create a consistent, repeatable, and secure environment deployment. Use the architectural guidance and reference implementation recommendations to accelerate the migration and deployment of RHEL-based workloads to Microsoft Azure.
- [Identity and access management \(IAM\) for RHEL](#): Discover IAM considerations for your RHEL landing zone accelerator deployment. Learn how to carefully design your hybrid cloud IAM implementation to ensure smooth integration and management of your instance landscape in the Azure cloud.
- [Business continuity and disaster recovery for RHEL](#): Learn how to improve business continuity and disaster recovery for your RHEL on Azure environment. Explore recommendations that you can use to support RHEL workloads and to deploy RHEL platform-management components.

- [Network topology and connectivity for RHEL](#): Learn how to implement design considerations and recommendations for network topology and connectivity in RHEL on Azure infrastructure. See how you can deploy various RHEL platform components and roles on virtual machines (VMs) with specific sizing and redundancy as needed.
- [Resource organization for RHEL](#): Learn key tactics for how to choose management groups and subscriptions that will help to ensure that you effectively govern and manage resources for your RHEL deployment.
- [Security for RHEL](#): See how you can design your security to target multiple areas to protect your RHEL systems. Learn how to create a secure and resilient cloud environment by implementing a strategic approach that applies both Azure and Red Hat security mechanisms.
- [Management and monitoring for RHEL](#): Learn about best practices for effective management and monitoring in your RHEL on Azure infrastructure.
- [Governance and compliance for RHEL](#): Learn about design considerations and recommendations for governance and compliance in an RHEL on Azure infrastructure. Discover key tactics for establishing efficient and effective governance and compliance in a cloud environment.
- [Platform automation for RHEL](#): Learn about the tools, features, and services you can use to automate various tasks and manage the RHEL lifecycle within your Azure environment. Discover how to implement automation to improve the efficiency and reliability of your RHEL on Azure infrastructure.

Oracle

- [Oracle on Azure IaaS landing zone accelerator](#): Learn how you can use the Oracle on Azure IaaS landing zone accelerator to automate the deployment of an environment capable of hosting Oracle on Azure IaaS Virtual Machines. See how the landing zone accelerator can be adapted to produce an architecture that fits your scenario and puts your organization on a path to sustainable scale.
- [Manage and monitor Oracle Database@Azure](#): Explore best practices for management and monitoring Oracle Exadata Database Service on a Dedicated Infrastructure with Oracle Database@Azure. Learn about key design considerations for health and metrics monitoring.
- [Business continuity and disaster recovery for Oracle Database@Azure](#): Learn about business continuity and disaster recovery for Oracle Database@Azure and how to build a resilient architecture for your workload environment. Discover how you can design your architecture to meet the recovery time objective (RTO) and recovery point objective (RPO) of your solution.

- [Business continuity and disaster recovery for Oracle on Azure Virtual Machines landing zone accelerator](#): Find significant updates that reflect new guidance including the deprecation of availability sets and new recommendations for Virtual Machine Scale Sets flexible orchestration.

Updated articles

- [Configure hybrid networking for Citrix on Azure](#): Find new guidance and additional recommendations for large scale deployments of Azure and Citrix Cloud environments in a single region.
- [Networking for Azure Virtual Desktop](#): Explore the new reference architecture for a hub and spoke topology with hybrid connectivity scenario.
- [Business continuity and disaster recovery for Azure Virtual Desktop](#): Learn about new resources for checking the zone resilience of your resources.

We made updates to the following articles to provide the latest guidance on networking:

- [Define network encryption requirements](#)
- [Plan for landing zone network segmentation](#)
- [Plan for traffic inspection](#)

These files were updated to include considerations for [Azure Arc-enabled VMware vSphere](#) and [Azure Arc-enabled System Center Virtual Machine Manager](#):

- [Hybrid and multicloud migration](#)
- [Ready methodology for hybrid and multicloud strategy](#)
- [Azure Policy machine configuration extension](#)

June 2024

New articles

- [Establish common subscription vending product lines](#): Give application teams the flexibility to deliver their workloads and services effectively by offering different subscription vending product lines. Implement subscription vending in your Azure landing zones to establish consistent scaling, security, and governance of Azure environments.

Updated articles

- [Ready methodology for hybrid and multicloud strategy](#): We updated this article to include Azure Arc-enabled VMware vSphere and Azure Arc-enabled System Center Virtual Machine Manager.
- [How to select a strategy for relocating cloud workloads](#): We updated the guidance on service and data-relocation automation methods.
- [Security, governance, and compliance disciplines for Azure VMware Solution](#): We updated this article to replace references to MMA, which is planned for deprecation. New guidance points to the Azure Monitor Agent.

We updated these articles to provide the latest guidance on Azure carbon optimization:

- [Sustainability considerations in cloud management](#)
- [Sustainability outcomes and benefits for business](#)

Hybrid/Azure Arc retirement

We retired several articles in the Hybrid/Azure Arc scenario in the best practices area. The content was outdated and no longer relevant to the Cloud Adoption Framework.

May 2024

New articles

This month, we introduced a new article related to Azure Virtual Network Manager that has recommendations for networking topologies in Azure landing zones. We also added new articles that have guidance on Oracle Database@Azure. Take a look at the new content to see how you can apply these recommendations in your organization.

Azure Virtual Network Manager

- [Azure Virtual Network Manager in Azure landing zones](#): Use Azure's Virtual Network Manager to implement landing zone design principles for application migrations, modernization, and innovation at scale. Learn more about two recommended networking topologies: Azure Virtual WAN and traditional hub-and-spoke. The Virtual Network Manager allows for the expansion and implementation of networking changes as business requirements evolve. See how these changes can be made without disrupting deployed Azure resources.

Oracle Database@Azure

Explore new articles on Oracle Database@Azure.

- [Introduction to the Oracle on Azure adoption scenario](#): Learn how to set up and manage Oracle workloads within your Azure landing zone. Learn about specific architectural strategies and implementations for Oracle database systems on Azure.
- [Identity and access management for Oracle Database@Azure](#): Learn key tactics for proper identity and access management for Oracle Database@Azure. Deploy your initial Oracle Database@Azure instance to create specific groups within Microsoft Entra ID and in the corresponding tenant. Learn how to use Microsoft Entra administrator groups and how to establish other groups and roles to enhance the granularity of access permissions.
- [Network topology and connectivity for Oracle on Azure Virtual Machines](#): Learn about network topology and connectivity considerations for running Oracle on Azure Virtual Machines. Explore the importance of security for Oracle workloads, and receive a high-level network design with various recommendations.
- [Network topology and connectivity for Oracle Database@Azure](#): Learn how to set up network topologies and connectivity for Oracle Database@Azure. Explore options for physical placement, learn about the use of virtual machine clusters, and learn the importance of private subnets. See how to configure network security groups and why you should use Azure Firewall to protect your Oracle Database@Azure instance.
- [Security guidelines for Oracle Database@Azure](#): Receive design considerations and recommendations for implementing security measures for Oracle Database@Azure. See the importance of a defense-in-depth strategy, which layers multiple defense mechanisms for comprehensive security. This strategy includes strong authentication and authorization frameworks, network security, and encryption of data.

Updated articles

Azure Blueprint deprecation

We made updates to reflect the deprecation of Azure Blueprint.

- [Govern antipatterns](#)
- [Resource consistency decision guide](#)
- [Get started: Document foundational alignment decisions](#)
- [Get started: Secure the enterprise environment](#)
- [Innovate methodology and maturity modeling](#)
- [Operational compliance in Azure](#)

- Understand the functions of a central IT team
- Function of cloud infrastructure and endpoint security
- Function of cloud security posture management
- Track costs across business units, environments, or projects
- Azure governance design area
- Inventory and visibility considerations
- Azure enterprise scaffold
- Tools and templates
- Data domains
- Self-serve data platforms
- Ready methodology for hybrid and multicloud strategy
- Balance competing priorities

Oracle Database@Azure updates

We updated articles to include guidance on Oracle Database@Azure.

- Capacity planning for Oracle on Azure
- Plan for Oracle on Azure adoption
- Strategic impact of Oracle on Azure
- Migration planning for Oracle on Azure

Azure landing zone multiregion updates

We updated articles to provide recommendations for multiregion deployments in Azure landing zones.

- Define an Azure network topology
- Resource naming and tagging decision guide
- Define your tagging strategy
- Traditional Azure networking topology
- Landing zone regions
- Hybrid identity with Active Directory and Microsoft Entra ID in Azure landing zones
- Management groups
- Subscription considerations and recommendations
- Resource organization design area overview

DevOps updates

- **DevOps considerations:** The DevOps technologies list was updated to include bootstrapping and infrastructure as code (IaC) tools.

April 2024

New articles

This month, we completely refreshed articles related to the Migrate and Govern methodologies in the Cloud Adoption Framework. We also added a few articles about Azure landing zones in the Ready methodology. Take a look to make sure you're applying the relevant recommendations.

Migrate methodology refresh

Explore dozens of new and updated articles to guide you through the migration process.

- **Prepare to migrate your workload**
 - [Migrate overview](#): Learn about the Migrate methodology and how to apply it as you move your workloads to Azure.
 - [Migration preparation checklist](#): Follow the checklist to plan for migration and to ensure that you have the right resources and tools in place.
 - [Prepare your landing zone for migration](#): Make sure you understand what you need to do after an Azure landing zone deployment to ensure that the technical environment supports migrations.
 - [Prepare tools and initial migration backlog](#): Prepare the tools and initial migration backlog that you need to support a migration to Azure.
 - [Select Azure regions for a migration](#): Choose the Azure regions that best meet your requirements for a migration.
 - [Align roles and responsibilities](#): Ensure clarity and coverage of essential functions for the migration to Azure by aligning roles and responsibilities across the departments in your organization.
 - [Get support resources and improve skills for migration projects](#): Make sure that your team has the right skills and resources to support a migration to Azure.
- **Assess your readiness**
 - [Migration assessment checklist](#): Follow the checklist to assess your workload's readiness for migration to the cloud.
 - [Classify workloads for a migration](#): Conduct a premigration assessment to classify your workloads based on the data sensitivity.
 - [Evaluate workload readiness](#): Understand how to adjust your workload to prepare it for migration to the cloud. Learn how to validate all assets and associated dependencies.

- [Design workload architecture before migration](#): Use the Cloud Adoption Framework to define the cloud architecture of a workload before you begin migration.
- **Prepare for deployment**
 - [Migration deployment checklist](#): Follow the checklist to prepare for the deploy phase of migrating a workload to Azure.
 - [Deploy supporting services for migration projects](#): Deploy the supporting services that your workloads need as you migrate them to Azure.
 - [Remediate assets prior to migration](#): Before the migration, remediate assets that you determine to be incompatible with your chosen cloud provider prior.
 - [Replicate assets in a cloud migration](#): Make sure that you understand the role of replication in the migration process and how to plan for the prerequisites and risks of replication activities.
 - [Prepare for management activities](#): Prepare to carry out management activities after a workload migration is complete.
 - [Test your migration deployment in Azure](#): Perform migration testing in Azure to ensure that your architecture works with the replicated or staged resources.
- **Complete a migration to Azure**
 - [Migration release checklist](#): Follow the checklist to release a workload to production after a migration to Azure.
 - [Change communication](#): Communicate changes to your organization before, during, and after a migration to Azure.
 - [Perform business testing during a migration](#): Perform business testing during a migration to ensure that your workloads are functioning as expected.
 - [Complete the migration to Azure](#): Complete final steps in the migration to Azure.
 - [Optimize cost after migration](#): Optimize costs to ensure that you're getting the most value from your cloud resources.
 - [Build a growth mindset by conducting retrospectives](#): Use retrospectives to build a growth mindset in your team and to improve your migration process.
- **Explore relevant migration scenarios**
 - [Review product migration scenarios](#): Review the migration scenarios that are available for your product.

Govern methodology refresh

We overhauled our collection of articles in the Govern methodology, making the concepts easier to consume and understand as you set up cloud governance in your organization.

- [Govern overview](#): Learn about the Govern methodology and follow the checklist to apply the methodology in your organization.
- [Improve landing zone governance](#): Improve the governance of your landing zones by following the best practices in this article.
- [Build a cloud governance team](#): Create a team that works to ensure the success of cloud governance in your organization.
- [Assess cloud risks](#): Assess, prioritize, and document cloud risks.
- [Document cloud governance policies](#): Define and document what should or shouldn't be done in your cloud environment.
- [Enforce cloud governance policies](#): Apply controls and procedures to align cloud use to the cloud governance policies.
- [Monitor cloud governance](#): Measure how well your cloud environment complies with your cloud governance policies.

Ready methodology

Find new articles about Azure landing zones in the Ready methodology.

- [Keep your Azure landing zone up to date](#): Make sure that your Azure landing zones are current to maintain improved security, avoid platform configuration drift, and stay optimized for new feature releases.
- [Migrate Azure landing zone custom policies to Azure built-in policies](#): Migrate your deprecated Azure landing zone custom policies to Azure built-in policies.
- [Update Azure landing zone custom policies](#): Update your Azure landing zone custom policies to ensure that they're current and compliant with the latest Azure policies.

Updated articles

- [Cloud adoption scenarios](#): Extensive revisions help you find scenarios that are relevant to your organization's cloud adoption journey.
- [Abbreviation recommendations for Azure resources](#): Find updates about the data-collection and alert-processing rules.
- [Zero Trust configuration for multitenant defense organizations](#): Review a new multitenant architecture diagram and information about Microsoft Entra ID Protection.
- [The Azure Well-Architected Framework for HPC](#): Explore updates related to ExpressRoute.
- [Manufacturing HPC storage in Azure](#): Learn about updates related to Azure Managed Lustre.

March 2024

Updated articles

- [Azure governance design area](#): Explore a new section for third-party tooling, including guidance for AzAdvertiser and Azure Governance Visualizer.
- [Tools and templates](#): Find information about governance for AzAdvertiser.
- [Resource consistency decision guide](#): Check out our expanded information about basic grouping for resource groups.
- [Select Azure regions](#): We added guidance about how to plan Azure resource group deployments.
- [Transition an existing Azure environment to the Azure landing zone conceptual architecture](#): Find tip to help you reduce the impact of regional outages.
- [Security guidelines for Oracle on Azure Virtual Machines landing zone accelerator](#): Review new use cases for centralized identity management. These use cases include using Azure Key Vault to store credentials and using hardened operating system images.
- [Storage for Azure HPC in the finance sector](#): Find new data to help you compare Azure Managed Lustre with Blob Storage, Azure Files, and Azure NetApp Files.
- [Network topology and connectivity for an SAP migration](#): Explore design recommendations for Azure ExpressRoute.

February 2024

New articles

- [Application identity and access management](#): Learn about recommendations that application owners and developers can use to design the identity and access management for cloud-native applications.

Updated articles

- [Hybrid identity with Active Directory and Microsoft Entra ID in Azure landing zones](#): We updated this article to include information about how to design and implement Microsoft Entra ID and hybrid identity for Azure landing zones. Microsoft Entra ID is a cloud-based identity and access management service that provides robust capabilities to manage users and groups. You can use it as a standalone identity solution or integrate it with a Microsoft Entra Domain Services

infrastructure or an on-premises Active Directory Domain Services (AD DS) infrastructure.

- [Landing zone identity and access management](#): Find out about considerations and recommendations for implementing identity and access control within Azure application and platform landing zones. This article has extensive new content.
- [Azure identity and access management design area](#): Learn about the identity and access management design area, which provides best practices to establish the foundation of your public cloud architecture. This article has extensive new content.
- [Cloud adoption journey](#): Learn about various types of cloud adoption journeys, including when to retire, replace, rearchitect, rebuild, rehost, or replatform your solution.
- [Azure migration tools decision guide](#): We added information about tools for application migration, modernization, replatforming, and rehosting.
- [Azure workload management and monitoring](#): Find new guidance about sovereign workloads.

January 2024

New articles

- [Define a sovereignty strategy](#): Organizations that use cloud services can find guidance for meeting the sovereignty requirements for their countries/regions. We also updated several articles with sovereignty considerations, which you'll find under the "Updated articles" section for this month.
- [Advanced Azure Policy management](#): Find out how to manage Azure Policy at scale by using the Enterprise Policy as Code (EPAC) open-source project and integrating IaC into your environment.

Updated articles

- [Plan for IP addressing](#): We added information about IPv6 considerations. Find recommendations to help you plan for IPv6 and implement it in your existing Azure networks.
- [Network topology and connectivity for Azure Arc-enabled servers](#): Find updates about how to overcome the management challenges involved in using Private Link.
- New sovereignty considerations can be found in the following updated articles:
 - [Select Azure regions for a migration](#)

- [Azure migration tools decision guide](#)
- [Select Azure regions](#)
- [Security design in Azure](#)
- We also added information about application migration tools and strategies. Find updates in these articles:
 - [Cloud adoption journey](#): Learn about rearchitecting or rebuilding applications that can't be replaced by SaaS or low-code solutions.
 - [Migration tools decision guide](#): Explore tools for application migration and modernization and tools for replatforming or rehosting.

December 2023

New articles

Find new guidance about Azure landing zones:

- [Manage application development environments in Azure landing zones](#)
- [Modify an Azure landing zone architecture to meet requirements across multiple locations](#)
- [Incorporate Zero Trust practices in your landing zone](#)

Updated articles

In the following articles, find updated guidance about workload discovery processes that help you understand the many dimensions involved in migrating a workload. You can use that information to help you effectively migrate cloud workloads to another region.

- [Evaluate a cloud workload for relocation](#)
- [Migrate a cloud workload to another region](#)
- [How to initiate a cloud relocation project](#)

In [Centralized security operations with external identities for multitenant defense organizations](#), we updated our guidance for centralized security operations.

In [Identity and access management for Azure Virtual Desktop](#), we added updates for Azure Virtual Desktop design considerations and supported identity scenarios.

Feedback

Was this page helpful?

 Yes

 No

Cloud adoption scenarios

Article • 08/01/2024

The following scenarios unify technical and nontechnical considerations that your organization can use to accelerate its overall cloud adoption journey.

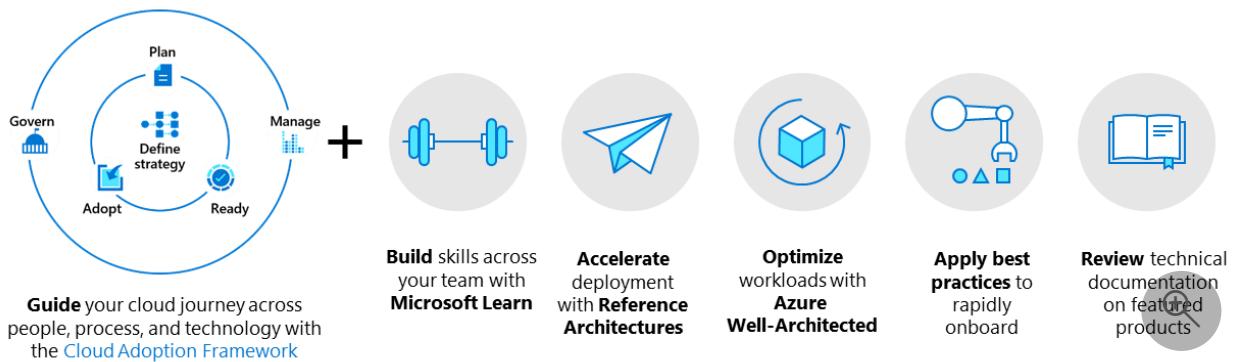
If you have a complex cloud adoption, you can follow trusted scenarios to narrow your focus to your requirements, which helps maximize your cloud investment. The scenarios include pointed guidance across several areas of cloud adoption best practices from the holistic and modular guidance of the Cloud Adoption Framework for Azure. This article also references workload-specific architectural guidance from the Azure Well-Architected Framework, which contains reference architectures and example scenarios with deployment templates to help you implement solutions. These solutions are based on extensive customer experiences that represent an infrastructure that your organization can rely on.

Learn and benefit from a repository of successful cloud adoption experiences and best practices. This guidance includes collected and codified successful migration and modernization experiences across Microsoft customer communities and partner communities. Incorporate best practices that suit your cloud adoption.

ⓘ Note

For migration guidance, see [Review product migration scenarios](#).

Components of a cloud adoption scenario



The following list describes the components of a cloud adoption scenario:

- [The Cloud Adoption Framework](#) guides you through each consideration and implementation along the phases of your cloud adoption journey. Use the Cloud

Adoption Framework across your organization to prepare decision makers, central IT, and the cloud center of excellence (CCoE) for your organization's cloud adoption efforts.

- [Microsoft Learn Training](#) is a free online training platform. Learn new skills and discover the power of Microsoft products with step-by-step guidance. These modules include role-based training and learning paths for Azure developers, solution architects, and administrators.
- **Reference architectures** are templates that include required components and technical requirements to implement the components. Each reference architecture includes best practices and considerations for scalability, availability, security, resilience, and other aspects of design. Reference architectures can help your organization accelerate deployment for many common scenarios.
- [The Well-Architected Framework](#) provides a set of Azure architecture best practices to optimize workloads and help you build and deliver great solutions. These best practices cover cost management, operational excellence, performance efficiency, reliability, and security. The Well-Architected Framework provides essential considerations for workload owners to review before they initiate their workload deployments.
- **Best practices** can help you build reliable, scalable, and secure applications in the cloud. Best practices provide guidelines and tips to implement efficient systems, mechanisms, and approaches.
- [Featured Azure products](#) provide information about the products that support your Azure strategy. Use proven combinations of Azure products and services to solve your business problems. Get started with documentation and reference architectures, follow best practices guidance for scenarios, and implement solutions for common workloads on Azure.

Scenarios to support your cloud adoption strategy

Each overview page provides guidance that you can use to implement the scenario in your cloud adoption strategy.

[] Expand table

Scenario	Description
Azure VMware Solution	Use an Azure service that's built with VMware to migrate your VMware resources to a vSphere-based, single-tenant, private cloud on dedicated infrastructure. Extend hybrid and multicloud agility. Learn how to optimize costs and develop cloud skilling so that you can minimize negative effects on business continuity and reduce the overall migration time.
Cloud-scale analytics	Ensure that your business data is discoverable, accurate, trusted, and protected. Scale complex data architectures within on-premises and multicloud scenarios without increased complexity. Enable common, self-serve data infrastructure, distributed architecture, secure network line of sight, and centralized data governance that's protected and reliable across deployments. You can process, deliver, and manage your mature data estate across deployments, migrate data platforms, democratize data, and extend workloads by using advanced cloud services.
Hybrid and multicloud	Confidently mature your analytics, AI, machine learning, and Internet of Things (IoT) capabilities in the cloud. Choose a unified operations approach to build, deploy, and migrate streamlined solutions across on-premises, multicloud, and edge environments. Modernize your existing on-premises investments, and gain the consistency and flexibility that you need to innovate across on-premises, multicloud, and edge environments. Manage and govern with a single control plane. Bring Azure services to any infrastructure. Extend compute storage to IoT devices, and run advanced machine learning analytics at the edge to gain real-time insights.
Modern application platform	Enable rapid innovation and workload portability with Kubernetes and container integration. Integrate platform as a service (PaaS) application services and containers into your strategy to maximize the value of cloud-enabled applications. Accelerate developer productivity by empowering developers to focus more on code and less on host environment concerns. Reduce operations costs with streamlined container orchestration and consistent runtimes. Modernize legacy workloads, and enable workload portability between container hosts across hybrid, multicloud, and edge environments with diverse container orchestration options. You can implement these changes before migration or modernization with customized application runtimes that are container host agnostic and meet legacy requirements in a cloud environment.
Oracle	Accomplish your business goals by using the Oracle on Azure framework to migrate and manage Oracle workloads to different technology platforms on Azure. Engage in parallel conversations across Oracle and central IT teams to align your cloud adoption plan. Separate discovery processes during strategic alignment with business leadership, and plan to prepare current-state and future-state data. Prepare your team and environment for migration by integrating Azure landing zones with preconfigured approaches to rapidly deploy environments to various technology platforms. Demonstrate how an Oracle migration can integrate with repeatable migration processes. Show how a common operations baseline can address run-state concerns during migration and meet the operational needs of other technology platforms.

Scenario	Description
SAP	<p>This scenario aligns with the Cloud Adoption Framework to help you migrate, innovate, and manage workloads and technology platforms and accomplish your business goals. Engage in parallel conversations across SAP and central IT teams to align your cloud adoption plan. Separate discovery processes into strategy during strategic alignment with business leadership, and plan to prepare current-state and future-state data.</p> <p>Prepare your team and environment for migration by integrating Azure landing zones with preconfigured approaches to rapidly deploy environments to various technology platforms. Demonstrate how an SAP migration can integrate with repeatable migration processes. Show how a common operations baseline can address run-state concerns during migration and meet the operational needs of other technology platforms. Integrate cloud-native solutions into your workloads.</p>
Virtual desktop	<p>Integrate Azure Virtual Desktop into your cloud adoption journey, and migrate your organization's user desktops to the cloud. Help improve employee productivity and accelerate the migration of various workloads that support the overall customer experience across your organization. Modernize existing virtual desktop environments, including session hosts, user profiles, images, and applications. Deploy a proven enterprise-scale architecture for Virtual Desktop to your current environment.</p> <p>Prepare for governance and operations at scale with enterprise-scale landing zones to ensure consistent governance, security, and operational controls across multiple landing zones. Use enterprise-scale landing zones to centralize the management of your virtual desktop environments. Use Azure-featured products to accelerate and improve virtual desktop capabilities.</p>
Retail industry	<p>Make modern retail resilient and competitive to provide cost savings, increase business agility, and heighten data security. You can unlock intelligent retail cloud maturity for your organization. The benefits include the ability to integrate cloud services across the supply chain with a PaaS, cloud-based web services development. You can create a deployment environment to ultimately deploy advanced software as a service (SaaS)-hosted AI and machine learning services. Use these services to implement actionable intelligence and develop intelligent applications.</p> <p>Identify future growth factors for your organization and forecast your customers' preferences. Gain hybrid capabilities and innovate in any environment by using cloud services that are delivered across on-premises, multicloud, or edge environments. Take advantage of modernized cloud services and resilient onsite computing. Use stores and distribution centers that always operate. These capabilities help you ensure security, privacy, compliance, and manageability within and among your environments.</p>
Defense	<p>This scenario provides universal guidance to help mission owners accelerate digital transformation. The guidance is impartial to country or region and includes best practices from defense organizations around the world.</p>

Scenario	Description
	<p>The defense cloud adoption scenario provides recommendations to help mission owners navigate unique challenges so that they can focus on meeting mission-specific objectives. You can find references to other documentation for deeper technical insights about critical areas. This scenario provides a secure, scalable, and governed environment that's tailored to mission needs.</p>

Next step

[Develop a cloud adoption strategy](#)

Feedback

Was this page helpful?

 Yes

 No

AI adoption

Article • 12/12/2024

This AI guidance provides a roadmap for [startups](#) and enterprises to adopt and maintain AI. It provides best practices. These best practices cover AI technology decisions (build vs. buy), skill development, team organization, and processes to govern, manage, and secure AI.

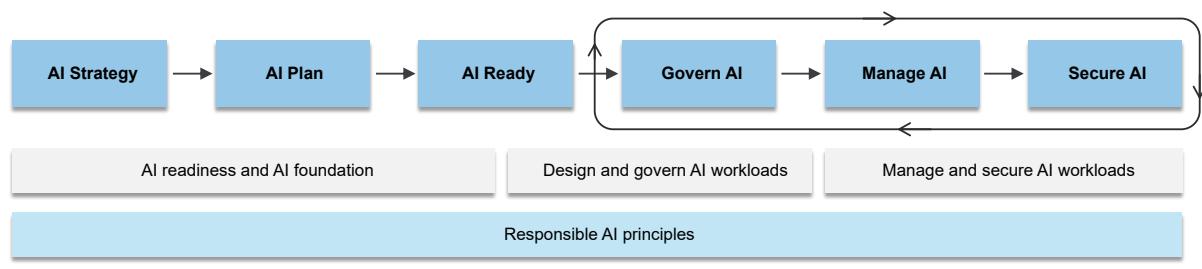


Figure 1. How to use the AI adoption guidance.

Why adopt AI?

AI frees you to focus on your top priorities. AI offers automation and autonomous agents that improve individual efficiency, boost productivity, and optimize a range of business processes.

How to adopt AI

Develop an [AI Strategy](#) to identify use cases and determine whether to build (PaaS and IaaS) or buy (SaaS) AI solutions for each use case. Develop an [AI Plan](#) to move AI use cases into production. If you **build** AI solutions with Azure PaaS or IaaS services, you must also get [AI Ready](#). **Buying** Microsoft Copilot SaaS solutions doesn't require this step. All AI adoption efforts must establish processes to [Govern AI](#), [Manage AI](#), and [Secure AI](#).

[+] Expand table

AI adoption step	Description	Applicable to build (PaaS or IaaS) or buy (SaaS)?
AI Strategy	Guidance to pick the right AI solutions.	For build (PaaS/IaaS) and buy (SaaS)
AI Plan	Guidance to execute AI adoption.	For build (PaaS/IaaS) and buy (SaaS)

AI adoption step	Description	Applicable to build (PaaS or IaaS) or buy (SaaS)?
AI Ready	Guidance to build AI workloads in Azure environment.	For build (PaaS/IaaS) only
Govern AI	Guidance to control AI.	For build (PaaS/IaaS) and buy (SaaS)
Manage AI	Guidance to maintain AI.	For build (PaaS/IaaS) and buy (SaaS)
Secure AI	Guidance to secure AI.	For build (PaaS/IaaS) and buy (SaaS)

AI checklists

Use the AI checklists as your roadmap for adopting and maintaining AI. The enterprise checklist prepares your organization to adopt AI at scale. The startup checklist helps you move toward production faster but still get governance, management, and security best practices.

[\[+\] Expand table](#)

AI adoption phase	Startup checklist	Enterprise checklist
AI Strategy	<input type="checkbox"/> Define an AI technology strategy	<input type="checkbox"/> Identify AI use cases <input type="checkbox"/> Define an AI technology strategy <input type="checkbox"/> Define an AI data strategy <input type="checkbox"/> Define a responsible AI strategy
AI Plan	<input type="checkbox"/> Access AI resources <input type="checkbox"/> Implement responsible AI	<input type="checkbox"/> Assess AI skills <input type="checkbox"/> Acquire AI skills <input type="checkbox"/> Access AI resources <input type="checkbox"/> Prioritize AI use cases <input type="checkbox"/> Create an AI proof of concept <input type="checkbox"/> Implement responsible AI <input type="checkbox"/> Estimate delivery timelines
AI Ready	<input type="checkbox"/> Build an AI environment <input type="checkbox"/> Choose an architecture <input type="checkbox"/> Use AI design areas	<input type="checkbox"/> Establish AI reliability <input type="checkbox"/> Establish AI governance <input type="checkbox"/> Establish AI networking <input type="checkbox"/> Establish an AI foundation <input type="checkbox"/> Choose an architecture <input type="checkbox"/> Use AI design areas
Govern AI	<input type="checkbox"/> Enforce AI governance policies	<input type="checkbox"/> Assess AI organizational risks <input type="checkbox"/> Document AI governance policies <input type="checkbox"/> Enforce AI policies <input type="checkbox"/> Monitor AI organizational risks

AI adoption phase	Startup checklist	Enterprise checklist
Manage AI	<input type="checkbox"/> Manage AI models <input type="checkbox"/> Manage AI costs	<input type="checkbox"/> Manage AI operations <input type="checkbox"/> Manage AI deployment <input type="checkbox"/> Manage AI endpoint sharing <input type="checkbox"/> Manage AI models <input type="checkbox"/> Manage AI costs <input type="checkbox"/> Manage AI data <input type="checkbox"/> Manage AI business continuity
Secure AI	<input type="checkbox"/> Implement AI security controls	<input type="checkbox"/> Assess AI security risks <input type="checkbox"/> Implement AI security controls <input type="checkbox"/> Maintain AI security controls

Next step

AI Strategy

Feedback

Was this page helpful?

 Yes

 No

Migrate to Azure with Azure VMware Solution

Article • 12/01/2022

Azure VMware Solution provides your organization with options when migrating to Azure. Migrating VMware resources from on-premises datacenters to a dedicated cloud environment on Azure can lower complexity, help you to minimize negative impacts to business continuity, and reduce the time required for your migration. Azure VMware Solution enables you to adopt cloud technology at a pace that matches your organization's requirements, adding cloud services incrementally as your business evolves.

Microsoft Azure VMware Solution

Azure VMware Solution is a first party Microsoft Azure service built with VMware that delivers a familiar vSphere-based, single-tenant, private cloud on Azure. The VMware technology stack includes vSphere, NSX-T Data Center, vSAN, and HCX. Running natively on dedicated infrastructure in Azure data centers, Azure VMware Solution provides a consistent experience to customers with existing on-premises VMware vSphere environments. Customers can set up the environment in hours, and quickly migrate virtual machine (VM) resources. Microsoft operates and supports the Azure VMware Solution environment and all necessary networking, storage, and management services.

Common customer journeys

- **Migrating VMware vSphere workloads to Azure:** the market reality is that organizations are moving to the cloud. The driving force for this migration timeline may be a need to quickly exit a data center, or scale your business capabilities beyond existing infrastructure. However, how your organization chooses to approach migration can vary significantly. For large businesses with existing VMware vSphere investments, it's important to maintain operational consistency for VMware vSphere environments in Azure.
- **Extending hybrid and multicloud agility:** customers seek common configurations like hybrid VMware architectures when adopting Azure VMware Solution. Organizations are looking to increase agility between resources on-premises and in the cloud by extending vSphere-based VMware environments to Azure. Hybrid and multicloud capabilities enable resource management consistency across on-

premises and cloud environments and modernize applications by connecting them to Azure-native cloud services.

- **Cost-optimizing licensing and billing:** in their migration journey, customers should consider how to best take advantage of on-premises licensing and billing to maximize their cloud investment. Microsoft offers Azure Hybrid Benefit, and Extended Security Updates for some versions of Windows and SQL Server. These extended security updates and licensing benefits are extended to Azure VMware Solution. With many VMware vSphere workloads now running Windows and SQL Server, customers can benefit from License Mobility - and reduce the economic impact of a cloud migration project. Centralized billing and support on Azure VMware Solution simplifies the tracking and management of cloud spend.
- **Developing cloud skilling:** Your organization's IT team has a great deal of expertise across on-premises systems like VMware vSphere. With the addition of cloud, many of these roles need retraining and more process improvements to adopt and manage new technologies. Developing these cloud competencies can be a significant investment in your time and resources. Because Azure VMware Solution provides consistency with how VMware vSphere resources are managed on-premises, there's little to no change required. Through a seamless connection to other Azure services, such as Azure Storage solutions, Azure hybrid identity, and monitoring and security services organizations can grow competencies incrementally in necessary cloud-focused up-skilling and cross-skilling.

Components of the scenario

This scenario is designed to guide the end-to-end customer journey, throughout the cloud adoption lifecycle. Completing the journey requires a few key guidance sets:

- **Cloud Adoption Framework:** These articles walk through the [considerations and recommendations of each CAF methodology](#). Use these articles to prepare decision makers, central IT, and the cloud center of excellence for adoption of Azure VMware Solution.
- **Reference architectures:** These [reference solutions](#) help to accelerate the deployment of Azure VMware Solution.
- **Featured Azure products:** Learn more about the [products that support your VMware vSphere strategy](#) in Azure.
- **Learn modules:** Get the [hands-on skilling required](#) to implement, maintain, and support Azure VMware Solution.

Next steps

Cloud adoption best practices encourage customers to create a single, centralized cloud adoption strategy by using the Strategy methodology of the Cloud Adoption Framework. The next article outlines technical considerations of Azure VMware Solution that might affect your strategy.

Strategic impact of Azure VMware Solution

Cloud-scale analytics

Article • 12/10/2024

With larger, more sophisticated forms of cloud adoption, your journey to the cloud becomes more complex. Azure cloud-scale analytics is a scalable, repeatable framework that meets your organization's unique needs for building modern data platforms.

Cloud-scale analytics covers both technical and nontechnical considerations for analytics and governance in the cloud. This guidance strives to support hybrid and multicloud adoption by being cloud agnostic, but the included technical implementation examples focus on Azure products.

Cloud-scale analytics has the following goals:

- Serve data as a product, rather than a byproduct
- Provide an ecosystem of data products, rather than a singular data warehouse that might not best fit your data scenario
- Drive a default approach to enforce data governance and security
- Drive teams to consistently prioritize business outcomes instead of focusing just on the underlying technology.

Cloud-scale analytics builds upon Microsoft's cloud adoption framework and requires an understanding of [landing zones](#). If you don't already have an implementation of Azure landing zones, consult your cloud teams about how to meet prerequisites. For more information, see [Ensure the environment is prepared for the cloud adoption plan](#).

Reference architectures allow you to begin with a small footprint and grow over time, adapting the scenario to your use cases.

Cloud-scale analytics includes repeatable templates that accelerate five core infrastructure and resource deployments. It's also adaptable for different organization sizes. If you're a small enterprise with limited resources, a centralized operations model mixed with some business subject matter experts might fit your situation. If you're a larger enterprise with autonomous business units (each with their own data engineers and analysts) as your goal, then a distributed operating model such as data mesh or data fabric might better address your needs.

Objectives

Cloud-scale analytics provides a framework that is built on the following principles. These principles address challenges with complex data architectures that don't scale to

the needs of organizations.

[+] Expand table

Principle	Description
Allow	<ul style="list-style-type: none">Scaling without increased complexitySeparation of concerns to facilitate governanceCreation of self-serve data infrastructure
Follow	<ul style="list-style-type: none">Best practices for well-architected cloud services
Support	<ul style="list-style-type: none">On-premises and multicloud scenarios
Adopt	<ul style="list-style-type: none">Product and vendor agnostic approachCloud Adoption Framework
Commit	<ul style="list-style-type: none">Azure landing zones as baseline infrastructure for all workloadsOperating model
Enable	<ul style="list-style-type: none">Common data infrastructureDistributed architecture under centralized governanceSecure network line-of-sight

Implementation guidance

Implementation guidance can be broken into two sections:

- Global guidance that applies to all workloads.
- Cloud-scale specific guidance

Global guidance

[+] Expand table

Documentation	Description
Cloud Adoption Framework	Managing and governing data is a lifecycle process, which begins by building on your existing cloud strategy and carries all the way through to your ongoing operations. The Cloud Adoption Framework helps guide your data estate's full lifecycle.

Documentation	Description
Azure Well-Architected Framework	Workload architecture and operations have a direct effect on data. Understand how your architecture can improve your management and governance of workload data.

Cloud-scale specific guidance

[+] Expand table

Section	Description
Build an Initial Strategy	How to build your data strategy and pivot to become a data driven organization.
Define your plan	How to develop a plan for cloud-scale analytics.
Prepare analytics estate	Overview of preparing your cloud-scale analytics estate with key design area considerations like enterprise enrollment, networking, identity and access management, policies, business continuity and disaster recovery.
Govern your analytics	Requirements to govern data, data catalog, lineage, master data management, data quality, data sharing agreements and metadata.
Secure your analytics estate	How to secure analytics estate with authentication and authorization, data privacy, and data access management.
Organize people and teams	How to organize effective operations, roles, teams, and team functions.
Manage your analytics estate	How to provision platform and observability for a scenario.

Architectures

This section addresses the details of physical implementations of cloud-scale analytics. It maps out the physical architectures of data management landing zones and data landing zones.

Cloud-scale analytics has two key architectural concepts:

- The data landing zone
- The data management landing zone

- Integration with software-as-a-service solutions such as Microsoft Fabric and Microsoft Purview

These architectures standardize best practices and minimize deployment bottlenecks for your development teams, and can accelerate the deployment of common cloud-scale analytics solutions. You can adopt their guidance for lakehouse and data mesh architectures. That guidance highlights the capabilities you need for a well-governed analytics platform that scales to your needs.

For more information, see: [Architectures Overview](#)

Best practices

The following advanced, level-300+ articles in the **cloud-scale analytics** table of contents can help central IT teams deploy tools and manage processes for data management and governance:

- [Data ingestion for cloud-scale analytics](#)
- [Data lake storage for cloud-scale analytics](#)
- [Use Azure Synapse Analytics for cloud-scale analytics](#)

Featured Azure products

Expand the **Featured Azure products** section in the **cloud-scale analytics** table of contents to learn about the Azure products that support cloud-scale analytics.

Common customer journeys

The following common customer journeys support cloud-scale analytics:

- **Prepare your environment.** Use the [Prepare your environment](#) articles as resources. Establish processes and approaches that support the entire portfolio of workloads across your data estate.
- **Influence changes to individual workloads.** As your cloud-scale analytics processes improve, your central data governance teams find requirements that depend on knowledge of the architecture behind individual workloads. Use the [Architecture](#) articles to understand how you can use the scenarios within for your use case.
- **Optimize individual workloads and workload teams.** Start with the [Azure Well-Architected Framework](#) guidance to integrate cloud-scale analytics strategies into

individual workloads. This guidance describes best practices and architectures that central IT and governance teams should use to accelerate individual workload development.

- **Use best practices to onboard individual assets.** Expand the **Best practices** section in the **cloud-scale analytics** table of contents to find articles about processes for onboarding your entire data estate into one cloud-scale analytics control plane.
- **Use specific Azure products.** Accelerate and improve your cloud-scale analytics capabilities by using the Azure products in the **Featured Azure products** section of the **cloud-scale analytics** table of contents.

Take action

For more information about planning for implementing the cloud-scale analytics, see:

- [Develop a plan for cloud-scale analytics](#)
- [Introduction to cloud-scale analytics](#)

Next steps

Begin your cloud-scale analytics journey:

[Introduction to cloud-scale analytics](#)

Feedback

Was this page helpful?

 Yes

 No

Introduction to the Azure high-performance computing (HPC) scenario

Article • 12/02/2022

This scenario focuses on modernizing your machine learning, visualization, and rendering workloads in Azure at any scale with HPC + AI through the Cloud Adoption Framework.

Defining HPC

High-performance computing (HPC) on Azure is the complete set of compute, networking, and storage resources integrated with workload orchestration services for applications that provide advanced analytics, graphic-intensive visualizations, and scalable rendering.

Industries such as automotive, energy, health and life sciences, silicon, finance, and manufacturing typically use Azure HPC workloads to run complex simulations such as 3D modeling or mathematical tasks with their known and familiar tools/processes to perform the intensive tasks with thousands of compute hours in a matter of days.

While the computation of their data is being orchestrated, each industry customer can focus more on analyzing and creating value from the results to prioritize and achieve their business goals in a cost effective and timely manner.

HPC narrative

As customers from different industries have specific needs that require HPC investment, there's also the concern of sustainability. Compared to deploying huge on-premises infrastructure, having your HPC workloads in the cloud helps to use minimal computing resources only as and when is needed with flexible scalability to support sustainability goals.

HPC in Azure also offers competitive pricing and performance compared to on-premises options, robust global regulatory compliance, and next-generation machine-learning tools to drive smarter simulations and empower intelligent decision making.

When customers choose the relevant Azure HPC solution for their specialized computing tasks, there will also need to be considerations for following the [Cloud Adoption Framework](#) to prepare decision makers and central IT for successful cloud adoption. See below for common customer HPC journeys.

Common Customer HPC Journeys

- Energy industry organizations such as oil and gas benefit from an up-scaled, end-to-end AI ecosystem that provides nurturing new ideas into safety operations and taking vast data repositories into AI solutions that promote operational advances such as predictive maintenance, simulation workloads, and automating completion reports to reduce data extraction times from months to hours.
- Finance organizations use HPC to modernizing approaches to risk management, portfolio optimization, and implementing effective compliance and governance practices to protect consumer data. Computationally intensive workloads in Azure HPC have reduced costs and increased/instantaneous scalability for on-demand pricing model calculations.
- Manufacturing organizations utilize HPC to fuel real-time product simulations that reduce time to market and improve product quality. Thousands of on-demand simulation jobs are executed to deliver faster solutions and more improved insights for following health and safety regulations.

Next step: Integrate HPC into your cloud adoption journey

Get started with the below list of guidance found at specific points throughout the cloud adoption journey to help you be successful in the cloud adoption scenario for your HPC environment.

- [Strategy for HPC](#)
- [Plan for HPC](#)
- [Review your environment or Azure Landing Zone\(s\)](#)
- [Migrate HPC](#)
- [Innovate with HPC](#)
- [Govern HPC](#)
- [Manage HPC](#)

For more information, refer to the [landing zone accelerator](#) for additional information.

[Design guidelines](#) are also available to provide guidance on creating landing zones for your HPC billing, identity and access management, network topology/connectivity, platform automation, management group and subscription organization, governance disciplines, and security disciplines.

Introduction to hybrid and multicloud

Article • 06/09/2023

Microsoft Azure provides all of the products and features required to help you build and operate your technology solutions in the cloud. We also understand that there are sound business reasons that may drive the necessity of using multiple private and public clouds. As a first step in your hybrid and multicloud journey, this article outlines and expands on Microsoft's unique perspective on important cloud computing terms.

Watch the following video to learn more.

<https://www.microsoft.com/en-us/videoplayer/embed/RWMBdw?postJs||Msg=true> ↗

Defining hybrid and multicloud

A hybrid cloud is a type of cloud computing that combines a private cloud (on-premises infrastructure), with a public cloud (computing services offered by third-party providers over the public internet). Hybrid clouds allow data and applications to consistently move between the two cloud environments. Many organizations choose a hybrid cloud strategy because of business requirements, such as meeting regulatory and data sovereignty requirements, maximizing on-premises technology investments, or addressing latency issues.

The hybrid cloud is evolving to include edge workloads. Cloud-managed edge computing devices bring the computing power of the public cloud to the private cloud, closer to where the IoT devices reside, including data residing in applications, connected devices, and mobile consumer services. Reducing latency by moving workloads to the edge, devices spend less time communicating with the cloud, and can operate reliably in extended offline periods. Expanded compute, storage, and service availability provides experience-driven resources closer to your customers.

Multicloud computing refers to the use of multiple cloud computing services from more than one cloud provider (including private and public clouds), in a heterogeneous environment. A multicloud strategy provides greater flexibility, and mitigates risk. Choose services from different cloud providers that are best suited for a specific task, or take advantage of services offered by a particular cloud provider in a specific location.

Hybrid and multicloud narrative

This scenario follows a common hybrid and multicloud narrative, and provides guidance on what you can do differently to be successful during your organization's cloud

adoption effort. This general narrative is not restricted to a single cloud adoption methodology, but takes a view of the entire cloud adoption journey.

A hybrid cloud platform gives your organization many advantages: greater flexibility, control, and scalability, with more deployment options, global scale, integrated cross-platform security, unified compliance, and improved workload, operational, and cost efficiencies across the enterprise, **consistently achieving more value from existing infrastructure**. When computing and processing demand fluctuates, hybrid cloud computing enables you to seamlessly scale up your on-premises infrastructure to the public cloud to handle any overflow, without giving third-party datacenters access to the entirety of your data. By running certain workloads in the cloud, your organization gains the flexibility and innovation the public cloud provides, while retaining highly sensitive data in your own datacenter to meet client needs, or remain in compliance with regulatory requirements.

This allows you to scale computing resources, while modernizing and protecting **mission-critical applications and data**. Eliminate the need to make massive capital expenditures to meet short-term spikes in demand, or being forced to free up local resources for more sensitive data. With cloud billing models, your organization will only pay for resources you temporarily use, instead of having to purchase, program, and maintain additional resources and equipment that could remain idle for long periods.

Another capital expenditure that could be eliminated is in offsite disaster recovery and backup infrastructure investments. Public cloud for BCDR strategies is a compelling option for those on-premises workloads and associated data not restricted in some way from residing in a public cloud. By using public cloud for BCDR customers take advantage of the major investments in privacy and security, scale on demand and ease and speed of recovery.

Companies are spreading resources across on-premises, multiple clouds, and the edge. Customers have four common needs that we often hear about:

1. Visibility into the health of all existing and future infrastructure and applications in a single pane of glass.
2. Difficulty integrating on-premises policies and updates with cloud infrastructure.
Organizations understand the need for implementing a governance standard,
3. A wide range of skills across on-premises and cloud, because there are often different application development teams in the organization. Customers are looking for consistent interoperability between the two so they can unify development practices.
4. Desire to manage security posture, without heavily modifying current operations.
Cloud and multicloud compounds this challenge, which can decrease trust and

increase apprehension.

Consider the deployment of cloud-native services in a hybrid and multicloud environment. Cloud services are often strictly contextualized as simply "moving data and applications to the public cloud". A hybrid strategy fully supports customer operations that preclude the use of the public cloud for some workloads, such as highly regulated industries like government infrastructure, healthcare, and financial services. Depending on geography and data sovereignty regulations, internal and customer data may be required to remain within the boundaries of on-premises datacenters. Data latency sensitivity requires compute to be close to source data in on-premises datacenters, and internet connectivity disruptions are expected, or have critical implications. In these scenarios, hybrid solutions that bring cloud services, decreased management overhead (maintaining these services on-premises), and a pay-as-you-go cloud billing model can be deployed in on-premises datacenters.

Hybrid and multicloud motivations

As a true enterprise-grade cloud provider, Azure supports your business objectives across public, hybrid, and multicloud environments. This series will discuss different best practices that can help facilitate various cloud mixes ranging from 100% Azure environments to environments that have little, or no, Azure infrastructure in place.

We recognize that there are many valid reasons for customers to choose to distribute their digital estate across hybrid and multicloud environments. Here are some common business drivers:

- Minimize or avoid single cloud provider lock-in
- Business units, subsidiaries, or acquired companies have already adopted different cloud platforms
- Different cloud providers may have regulatory and data sovereignty requirements in different geopolitical regions
- Improve business continuity and disaster recovery by duplicating workloads across two cloud providers
- Maximize performance by running applications close to user locations, which may require hybrid or multicloud adoption
- Enable easy migration for some data platforms or industry-specific applications by adopting multicloud strategies

Hybrid and multicloud concerns

Some of the motivations listed above can become business transformations with a sound hybrid and multicloud adoption strategy.

Others require significant effort predeployment and post-deployment efforts to realize those innovative benefits. Cloud provider lock-in, for instance, is possible. But to avoid lock-in, organizations are required to limit their vision for cloud adoption. Many of the most beneficial products and features in a cloud provider are not portable to other cloud providers. To achieve portability and minimize lock-in, organizations are often required to limit cloud adoption to basic infrastructure as a service (IaaS) capabilities, or invest heavily in the use of cloud-native technologies like containers or Kubernetes.

After workloads are released and are in production, another common concern associated with hybrid and multicloud adoption surfaces: when organizations attempt to provide operations management support to workloads in new environments, they often have to quickly rethink their practices. Existing operations management platforms (including existing operations management policies and processes), were not built for these types of environments. To account for deviations in cloud environments, companies often end up with disparate operations tooling and operations practices, which multiplies the cost of operations by the number of cloud environments supported.

Next step: Minimize hybrid and multicloud concerns with unified operations

Understand the concept of unified operations before starting your hybrid and multicloud journey; consistent operations practices across all of your cloud environments with a common control plane can help to address many concerns regarding hybrid and multicloud strategies.

Determine whether you need to duplicate operations for each cloud provider or implement a [unified operations approach to cloud management](#) before proceeding with hybrid and multicloud adoption at scale.

Introduction to the modern application platform scenario

Article • 12/01/2022

As customers address larger, more sophisticated forms of cloud adoption, their journey to the cloud becomes more complex. Commonly, customers use orchestrated containers to manage pools of workloads in one or more centralized clusters. This article series combines technical and non-technical considerations required to prepare for Kubernetes and container integration into centralized operations and your broader cloud strategy.

Organizations include managed services, application services, and containers in their overall strategy to accelerate developer productivity, reduce operating overhead, make workloads more portable, and modernize legacy workloads.

- **Application platform:** A collection of application dependencies designed to support the execution of the application. Application platforms accelerate development by providing well-defined structures to address many common needs. Application platforms can also create constraints that affect how the application can be built.
- **Application services (PaaS services):** Application services, or platform as a service (PaaS) options for developers, provide an application platform that maximizes the value each application can draw from a cloud environment. These services create a layer of abstraction between the application platform and the underlying cloud infrastructure. This layer of abstraction forces a specific and consistently defined set of application platform constraints, which can only run in specific environments.
- **Containers:** Containers create a similar layer of abstraction between an application's runtime requirements and the underlying operating system and infrastructure. Unlike PaaS options, containers allow the application runtime to be configured for the needs of the application, regardless of the container host.
- **Container orchestration:** A container orchestrator provides a consistent application runtime for an instance of a container host. This orchestration allows workload-focused operations teams to mature deployment and DevOps practices to facilitate multiple deployment and operations practices, reducing dependencies on centralized or human operations.

This article series will outline how application services and containers can be integrated into your cloud adoption strategy to delivery the following outcomes:

- **Developer acceleration through abstraction:** Both containers and application services, accelerate developer productivity by allowing developers to focus more on code and less on host environment concerns.
- **Reduce operations costs through abstraction:** Standardized container orchestration ensures consistent runtimes across all hosts, which streamlines operations regardless of the workloads developers may deploy. Through standardization, centralized operations teams can extend traditional support and operations practices to container hosts.
- **Workload portability through abstraction:** Moving workloads between container orchestrators allows for workload portability. Some container orchestration platforms work in the public cloud. Other container orchestration platforms are designed for edge or private cloud deployment. Allowing for diverse container orchestration options allows for portability of workloads between container hosts across hybrid, multicloud, edge, and public cloud platforms.
- **Modernize legacy workloads through abstraction:** Legacy workloads may require a layer of abstraction prior to migration or modernization. Customizing the runtime on a container host allows the legacy requirements to be met in a modern cloud environment.

This article series outlines how you can integrate container and container management into your strategy, plan, adoption, and operation phases of your cloud journey.

Components of the scenario

This scenario is designed to guide the end-to-end customer journey, throughout the cloud adoption lifecycle. Completing the journey requires a few key guidance sets:

- **Cloud Adoption Framework:** These articles walk through the smallest set of considerations and implementations of each CAF methodology. Use these articles to prepare decision makers, central IT, and the cloud center of excellence for adoption of containers and container management as a central part of your technology strategy.
- **Azure Well-Architected Framework:** These articles outline the considerations that each workload owner should make when their workloads need to be deployed using containers or container management solutions like Kubernetes.
- **Reference architectures:** These reference solutions aid in accelerating deployment of container solutions using Azure Kubernetes Service (AKS).
- **Featured Azure products:** Learn more about the products that support your container and container management strategy in Azure.
- **Learn modules:** Gain the hands-on skills required to implement, maintain, and support container and AKS solutions.

Common customer journeys

AKS reference architectures: The reference architectures listed in the left pane demonstrate how to deploy various proven architectures to manage your container and Kubernetes platforms with the help of Azure Kubernetes Service (AKS). These architectures are the suggested starting point for Kubernetes in Azure.

Migrate existing workloads to AKS: A common use case for AKS in Azure is to modernize existing web-based workloads directly to a container-based or cloud-native solution, instead of traditional migration efforts. The article on [migrating to containers](#) will demonstrate how Azure Migrate can accelerate container migration within your standard migration processes.

Centralize deployment and management of containers: The first set of articles in the left pane provides rich guidance on centralization of your container strategy. This article series intends to help central IT or cloud center of excellence teams understand how containers affect your cloud strategy and how to provide consistent centralized support.

Prepare for governance and operation of containers at scale: The [AKS landing zone accelerator](#) demonstrates how you can use enterprise-scale landing zones to ensure consistent governance, security, and operations across multiple landing zones for centralized management of containers at scale.

Implement specific Azure products: Accelerate and improve container and Kubernetes capabilities using different kinds of Azure products outlined in the featured products section.

Next step: Integrate modern application platforms into your cloud adoption journey

The following list of articles will take you to guidance at specific points in the cloud adoption journey to help you be successful in the cloud adoption scenario.

- [Strategy for modern application platforms](#)
- [Plan for modern application platforms](#)
- [Review your environment or Azure landing zones](#)
- [Migrate workloads to modern application platforms](#)
- [Innovate using modern application platform solutions](#)
- [Govern modern application platform solutions](#)
- [Manage modern application platform solutions](#)

Introduction to Oracle on Azure adoption scenarios

Article • 11/07/2024

This article describes how to set up and manage Oracle workloads within your Azure landing zone. The architectures described incorporate a multi-region design. The article also describes specific architectural strategies and provides reference implementations for Oracle database systems that cross multiple regions on Azure. The guidance assumes that you have an Azure landing zone that's configured to support multi-region deployments. For more information, see [What is an Azure landing zone?](#) and [Landing zone implementation options](#).

Define Oracle on Azure

Oracle on Azure adoption scenarios provide two principal technology platform options:

- **Oracle on Azure Virtual Machines:** Run Oracle databases and enterprise applications, such as Siebel, PeopleSoft, JD Edwards, E-Business Suite, or customized WebLogic Server applications on Azure infrastructure. You can use an Oracle Linux image, Red Hat Enterprise Linux (RHEL), or another endorsed operating system. There are multiple VMs and storage options available.
- **Oracle Database@Azure:** You can use Oracle Database@Azure to run Oracle Exadata infrastructure in Azure. Oracle Exadata is a high-performance database platform. Oracle Database@Azure supports tools, such as Oracle Real Application Clusters (RAC) and Oracle Data Guard. Oracle enterprise applications such as Siebel, PeopleSoft, JD Edwards, E-Business Suite, or customized WebLogic Server applications run on Azure VMs and can connect to Oracle Database@Azure.

For more information, see [Oracle on Azure overview](#).

Oracle on Azure architectures

You can use Oracle on Azure to integrate database services for Oracle with Azure cloud capabilities. Apply data analytics and generative AI to your Oracle data. Monitor your apps and Oracle database service with a single view in Azure.

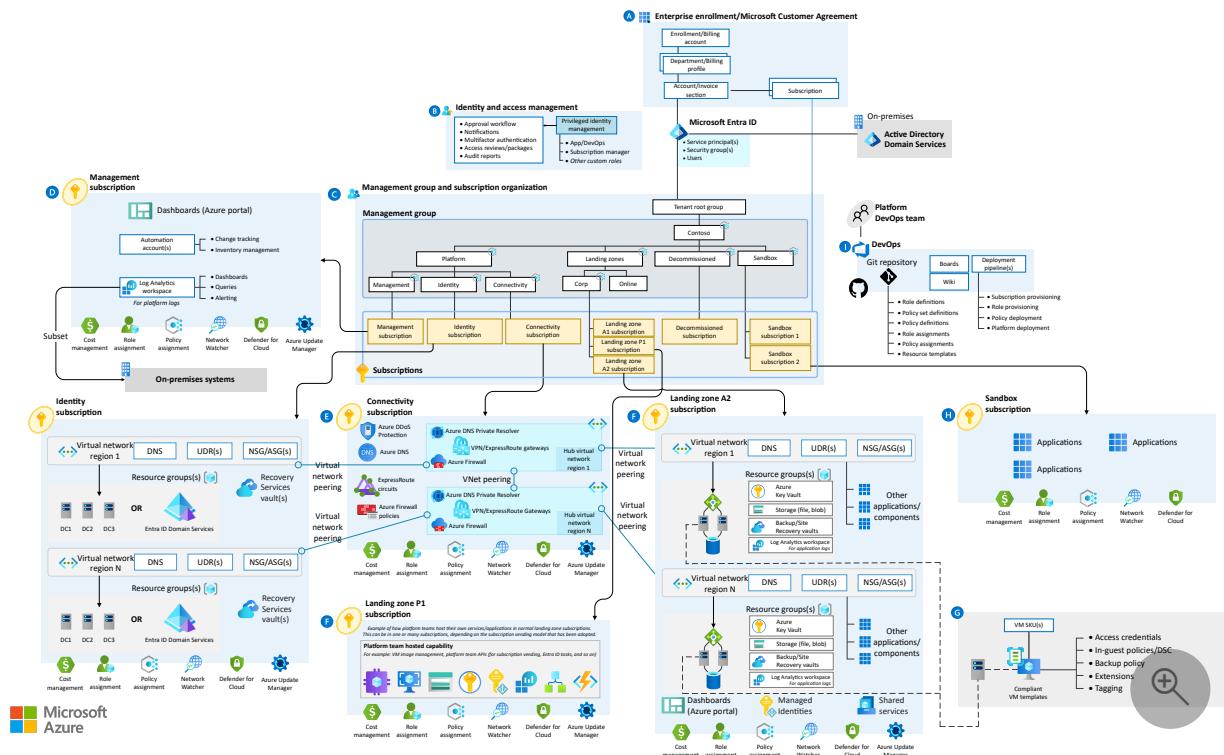
The following diagrams provide a conceptual reference architecture for Oracle on Azure Virtual Machines and Oracle Database@Azure that you can use to accelerate cloud adoption. The diagrams show critical design areas for Azure landing zones and

incorporate a multi-region design to enhance availability and disaster recovery. They also present a network layout that demonstrates architectural principles across multiple regions, but they don't detail an entire enterprise network.

Use the multi-region reference architectures as a starting point. Modify the reference architectures to fit your specific business and technical requirements when you plan to implement your landing zone. Implementing a multi-region architecture can help you ensure business continuity and resilience against regional outages. It aligns with best practices for high availability and scalability.

Landing zone architecture for Oracle on Azure virtual machines

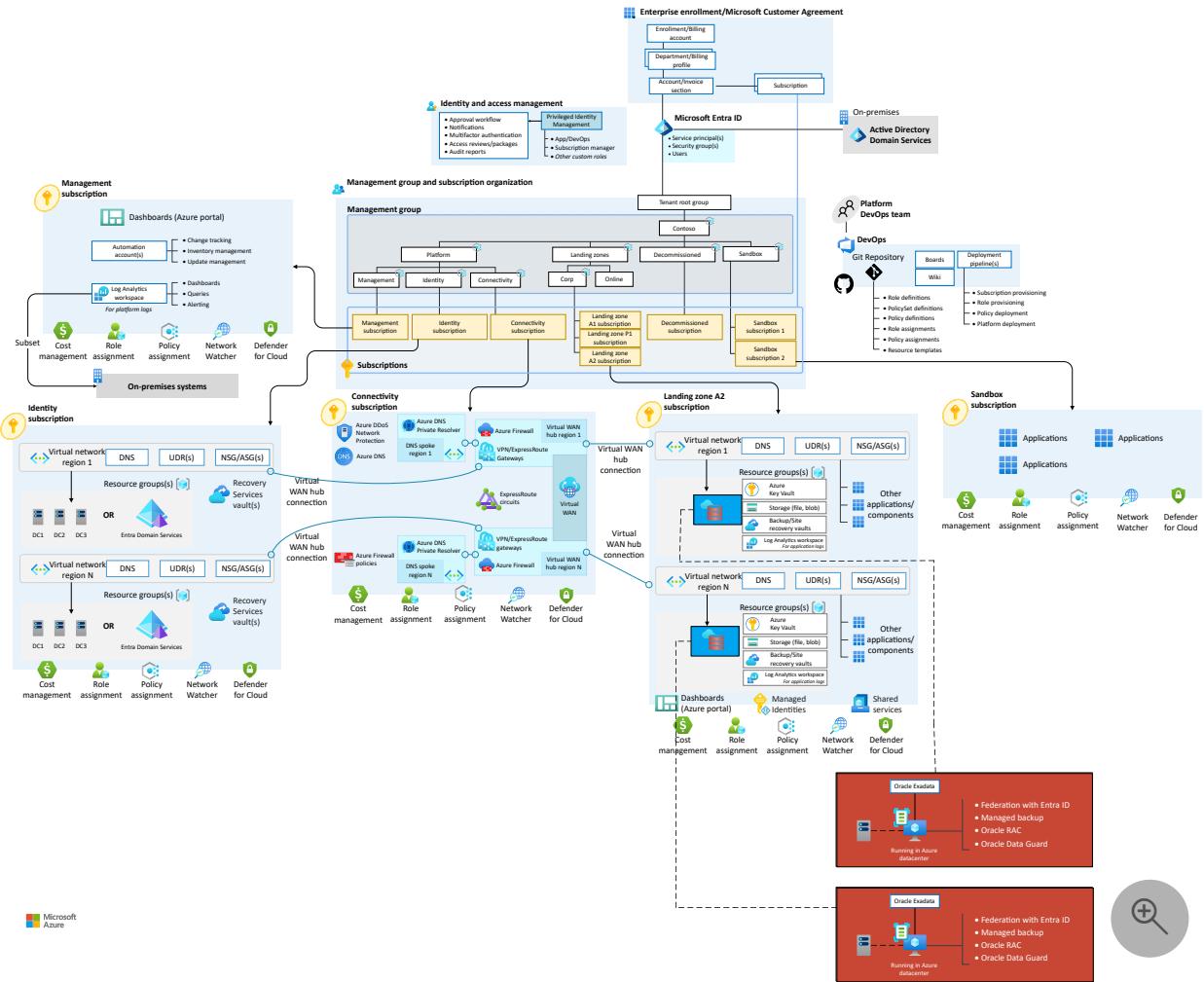
The following diagram shows Oracle on Azure virtual machines deployed to an application landing zone in a multi-region design. The approach shown distributes your Oracle databases across multiple Azure regions to enhance availability, scalability, and disaster recovery capabilities. The diagram also highlights the critical design areas that support this multi-region deployment. The Oracle databases run on VMs in each region. You can change the number and size of VMs to accommodate your needs.



Landing zone architecture for Oracle Database@Azure

The following diagram shows Oracle Database@Azure deployed to an application landing zone in a multi-region design. It also shows the critical design areas that support this multi-region deployment. The Oracle databases run on Oracle Exadata VM

clusters across multiple regions on Azure. You can change the number and size of VM clusters in each region to accommodate your needs.



Next step

Strategic impact of Oracle on Azure

Feedback

Was this page helpful?

Yes

No

Introduction to an SAP adoption scenario

Article • 12/01/2022

This article series outlines the process for integrating an SAP platform into your cloud adoption efforts.

Executive summary of SAP on Azure

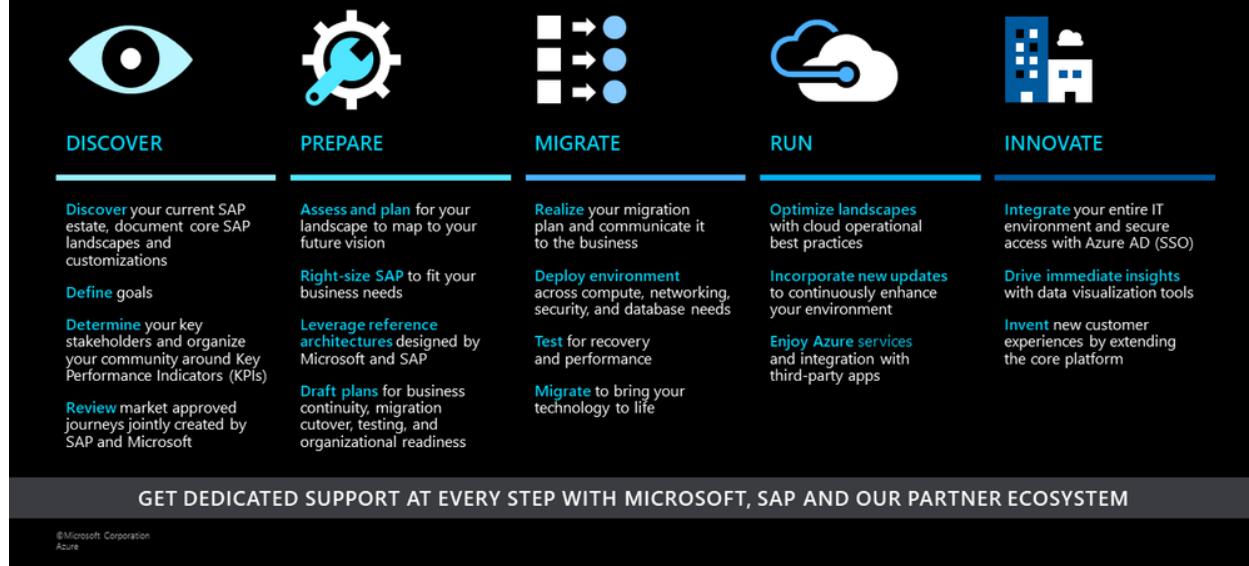
These articles describe how SAP workloads impact your overall strategy, cloud adoption plan, and environmental readiness efforts, with detailed guidance on common drift for each effort. Once an environment is established, the series explains the processes for migrating an SAP platform and how to use cloud technologies to innovate on that platform. To support your cloud adoption needs, the series also outlines considerations and best practices for managing governance and operations throughout an SAP deployment.

To accelerate these efforts, the articles also include detailed technical resources that describe how to build an enterprise-scale landing zone that can support your mission-critical SAP needs.

How to align to the SAP migration framework

The SAP on Azure team has produced a comprehensive framework for migrating and innovating with SAP on Azure. That framework is the main guide for organizations that focus on delivering the most successful SAP migration to Azure. If your highest priority is successfully migrating SAP to the cloud, then continue to use that guide.

Migrate your SAP landscape to Microsoft Azure



This article series outlines an SAP scenario for the Cloud Adoption Framework, and it complements the current SAP migration framework. These articles are also designed for a slightly different audience with slightly different goals, and they can help your organization if you're integrating the migration of SAP into your overall cloud adoption plan. As organizations migrate to the cloud, they typically need to migrate, innovate, and manage a range of workloads and technology platforms to accomplish their business goals. Those efforts align with the methodologies in the Cloud Adoption Framework for consistent processes and approaches across various technology platforms and workloads.

If your team is following the guidance in both frameworks, you'll see very similar guidance just in packaging that better aligns with the audience and their objectives. Below is a list of terms for the methodologies that can help both audiences to have a similar conversation:

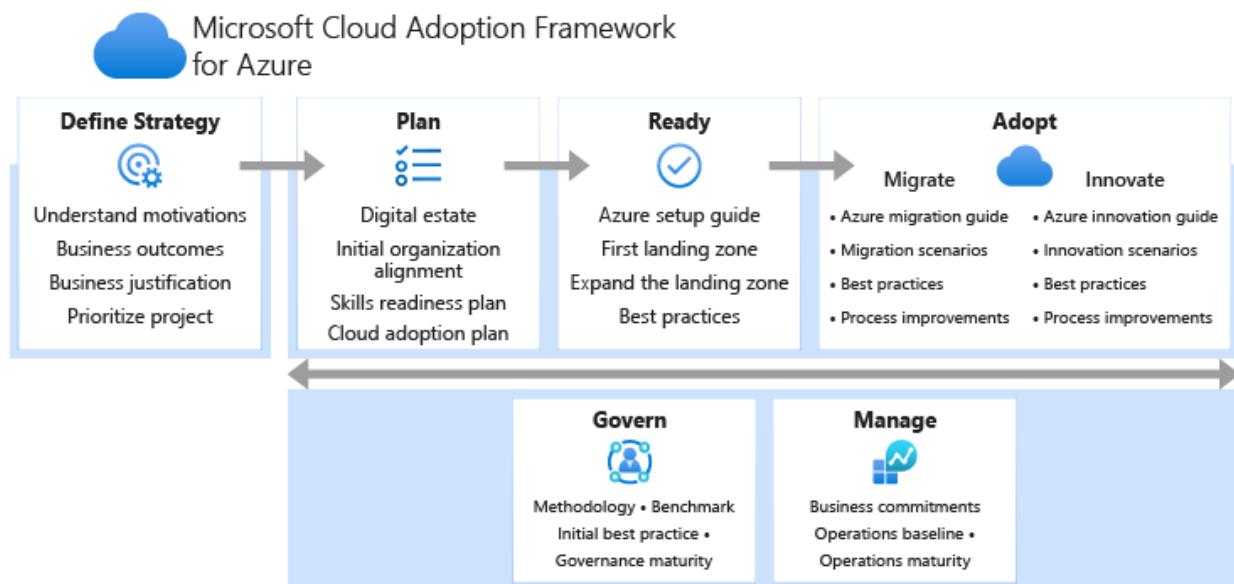
Discover	When cloud adoption spans multiple technology platforms, it's helpful to separate the discovery process into two conversations: Strategy , to engage business executives during strategic alignment, and Plan , to prepare the plan based on current and future-state data.
Prepare	The Ready methodology methodology prepares your team and the environment for the coming work. This scenario also integrates Azure landing zones to provide prebuilt and opinionated approaches to help your team to rapidly deploy environments that can support various technology platforms.
Migrate	The Migrate methodology demonstrates how an SAP migration can integrate with other repeatable migration processes.

Run	The Manage methodology shows how a common operations baseline can address many of the run-state concerns addressed in Migrate and also meet the operational needs of other technology platforms.
Innovate	The Innovate methodology outlines how you can take your SAP platform to the next level and integrate cloud-native solutions into your workloads.

The guidance aligns these sources to help your SAP and central IT teams work together during all phases of cloud adoption.

The SAP adoption process

Adopting SAP workloads in the cloud incorporates and addresses most of the methodologies and phases within the Cloud Adoption Framework. Distinct constraints within each phase will require actions specific to SAP, and this article series maps standard processes to SAP-specific tasks.



Next steps

The following articles provide guidance for specific points throughout the cloud adoption journey to help you succeed in adopting SAP in Azure.

- [Strategic impact of SAP in the cloud](#)
- [Plan for SAP cloud adoption in Azure](#)
- [Review your environment or Azure landing zones](#)
- [Migrate an SAP platform to Azure](#)
- [Innovate with SAP](#)

- Manage SAP

Migrate end-user desktops to Azure Virtual Desktop

Article • 03/07/2024

Migrating an organization's end-user desktops to the cloud is a common scenario in cloud migrations. Doing so helps improve employee productivity and accelerate the migration of various workloads to support the organization's user experience.

Components of the scenario

This scenario is designed to guide the end-to-end customer journey, throughout the cloud adoption lifecycle. Completing the journey requires a few key guidance sets:

- **Cloud Adoption Framework:** These articles walk through the considerations and recommendations of each CAF methodology. Use these articles to prepare decision makers, central IT, and the cloud center of excellence for adoption of Azure Virtual Desktop as a central part of your technology strategy.
- **Reference architectures:** These reference solutions aid in accelerating deployment of Azure Virtual Desktop.
- **Featured Azure products:** Learn more about the products that support your virtual desktop strategy in Azure.
- **Training modules:** Gain the hands-on skills required to implement, maintain, and support an Azure Virtual Desktop environment.

Common customer journeys

- **Azure Virtual Desktop reference architecture:** The [Azure Virtual Desktop reference architecture](#) demonstrates how to deploy a proven architecture for Azure Virtual Desktop in your environment. This architecture is a suggested starting point for Azure Virtual Desktop.
- **Migrate existing virtual desktops to Azure:** A common use case for Azure Virtual Desktop is to modernize an existing virtual desktop environment. While the process can vary, there are several components to a successful migration, like session hosts, user profiles, images, and applications. If you're migrating existing VMs, you can review articles on migration to learn how tools like [Azure Migrate](#) can speed up your migration as part of a standard migration process. However, your migration might consist of bringing your golden image into Azure and provisioning a new Azure Virtual Desktop host pool with new session hosts. You

can migrate your existing user profiles into Azure and build new host pools and session hosts as well. A final migration scenario might include migrating your applications into MSIX app attach format. For all of these migration scenarios, you need to provision a new host pool because there's currently no direct migration of other virtual desktop infrastructure (VDI) solutions into Azure Virtual Desktop.

- **Prepare for governance and operations at scale:** Enterprise-scale support for Azure Virtual Desktop demonstrates how you can use enterprise-scale landing zones to ensure consistent governance, security, and operations across multiple landing zones for centralized management of virtual desktop environments.
- **Implement specific Azure products:** Accelerate and improve virtual desktop capabilities using different Azure products outlined in the featured products section.

Next steps

The following list of articles will take you to guidance at specific points in the cloud adoption journey to help you be successful in the cloud adoption scenario.

- [Strategy for Azure Virtual Desktop](#)
- [Plan for Azure Virtual Desktop](#)
- [Migrate to Azure Virtual Desktop](#)
- [Manage an Azure Virtual Desktop environment](#)
- [Govern an Azure Virtual Desktop environment](#)

Feedback

Was this page helpful?



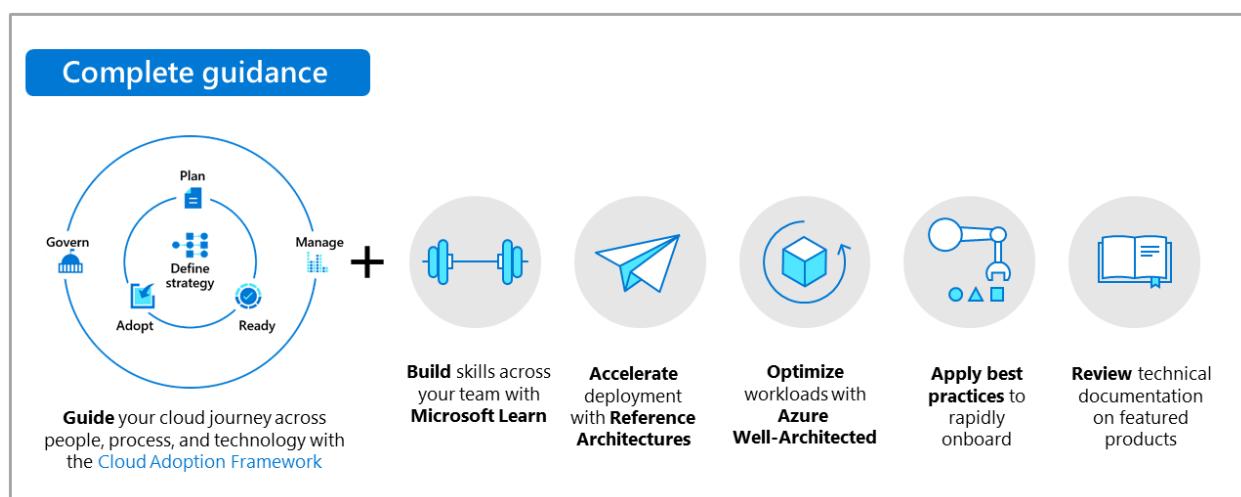
Cloud adoption for the retail industry

Article • 12/01/2022

As retail companies address larger and more sophisticated forms of cloud adoption, their journey to the cloud becomes more complex. This series of articles guides you in planning a retail cloud adoption journey.

Scenario components

This scenario is designed to guide a retail company's cloud adoption journey from beginning to end.



Use the following information to guide you on your journey:

- ④ **Microsoft Cloud Adoption Framework for Azure**: These articles provide guidance for planning a cloud adoption journey. They are the right starting point for business decision makers, technology leaders, solution architects, and infrastructure architects.
- ④ **Microsoft Azure Well-Architected Framework**: These articles outline what every retail workload owner should consider when deploying and managing workloads in the cloud.
- ↗ **Retail solutions**: These architecture solutions are developed and recommended for common retail industry scenarios. You can customize them to fit your business needs and accelerate deployment.
- ↔ **Learn modules**: Learn the skills to implement, maintain, and support your solution.

Common customer challenges and supporting guidance

Prepare for centralized operations or cloud center of excellence: Review the [Cloud Adoption Framework](#) articles. Establish the processes and approaches required to support your workloads.

Monitor assets across an existing retail portfolio: Focus on the govern and manage articles to integrate Azure into your existing operations processes. Use the ready article to deploy those improvements across all of your cloud environments.

Influence changes to individual workloads (Central IT/Cloud Center of Excellence (CCoE)): As controls improve, the central IT teams will need to know the architectures that handle the workloads. Use the [Azure Well-Architected Framework](#) guidance to help workload owners improve the handling of the workloads.

Optimize individual workloads (workload teams): Workload owners should start with the [Microsoft Azure Well-Architected Review](#) guidance to understand the best ways to handle their workloads to support company business strategies. A central IT or CCoE team that supports the workload can use the guidance for insights into best practices and architectures to accelerate implementation.

Implement specific Azure products: Use Azure products outlined in the featured products section to improve and accelerate your implementation.

Next steps

The following articles can guide your cloud adoption journey and help you succeed in the cloud adoption scenario for the retail industry.

- [Cloud adoption strategy for the retail industry](#)
- [Cloud adoption plan for the retail industry](#)
- [Review your environment or Azure landing zones](#)
- [Migrate common retail industry technologies](#)
- [Innovation in the retail industry](#)
- [Governance in the retail industry](#)
- [Management in the retail industry](#)

Introduction to the defense scenario

Article • 06/19/2023

The CAF Defense scenario is cloud adoption guidance tailored for defense organizations. Defense organizations coordinate and supervise armed forces in the interest of national/regional security. Data must be current, secure, and available from the tactical edge to headquarters distributed around the globe. The defense scenario addresses the challenges mission owners face in cloud adoption and offers recommendations to navigate those challenges. The goal is to help accelerate the digital transformation efforts of defense organizations. The defense scenario is divided into the command, platform, and mission domains.

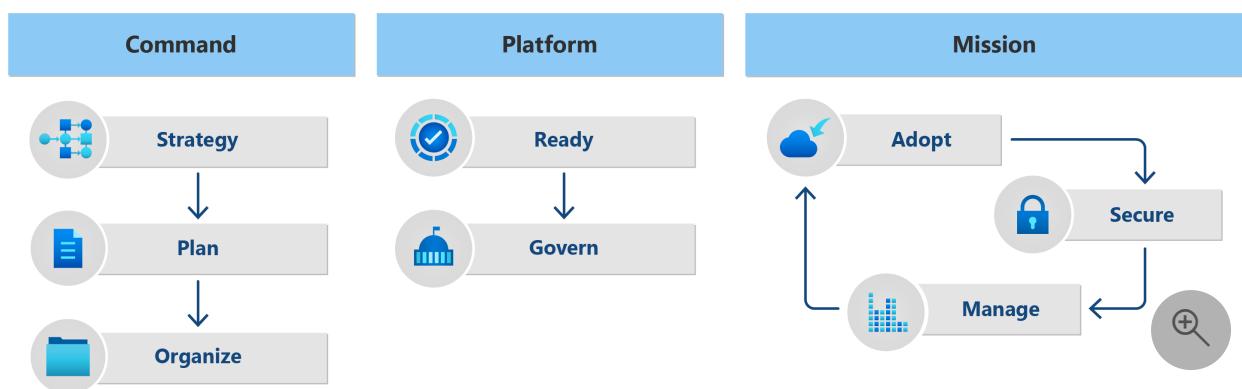


Figure 1: Overview of cloud adoption domains and methodologies

Command domain

The command domain focuses on preparing defense organizations for cloud adoption so that they can achieve their objectives. Here mission owners identify goals and motivators and define a roadmap to achieve those identified goals. They also involve the right people to develop, execute, and maintain the cloud solution.

Platform domain

The platform domain focuses on establishing a foundation for Mission applications and services to run in the cloud. The high priority in this domain is aligning with a cloud broker approach. The preferred way to build a cloud platform is with a cloud broker. A cloud broker builds and maintains the platform while establishing connectivity, identity, and governance for the platform. The platform domain provides guidance on how to prepare and execute with a cloud broker.

Mission domain

The mission domain focuses on the iterative process of developing cloud solutions. The domain covers technology adoption and how to use those technologies to secure and manage defense workloads. Whether migrating or rearchitecting an application using cloud native services, architectural decisions need to be made that align with mission objectives, and the workload domain provides guidance in this area.

Next step

The defense scenario helps ensure defense organizations reach the objective in each domain of your cloud adoption journey. The command domain is the starting point of the defense scenario. It starts with building a strategy to meet broader defense initiatives.

Strategy

Get started with the Cloud Adoption Framework

Article • 12/12/2024

The Cloud Adoption Framework can help you get started with several different getting started guides. This article groups the guides to help you find the one that best aligns with your current challenges.

Each of the following links takes you to questions that are typically asked when an organization is trying to accomplish a certain goal during their cloud adoption journey.

- [Choose the cloud adoption scenario that best supports your strategy](#)
- [Examine antipatterns across methodologies and their solutions](#)
- [Align foundational concepts to onboard a person, project, or team](#)
- [Adopt the cloud to deliver business and technical outcomes sooner](#)
- [Improve controls to ensure proper operations of the cloud](#)

Cloud adoption scenarios

Your organization's cloud adoption effort should support long-term strategic goals for your cloud journey. We have cloud adoption guidance for different scenarios including hybrid and multicloud, modern application platforms, SAP, retail, defense, and more.

- [Scenarios overview](#)
- [Hybrid and multicloud](#)
- [Modern application platform](#)
- [SAP](#)
- [Desktop virtualization](#)
- [Retail industry](#)
- [Defense](#)

Cloud adoption antipatterns

You might encounter missteps with design, planning, or implementation when migrating to the cloud. We have updated detailed guidance on [antipatterns](#) that can block innovation and prevent businesses from adopting and realizing goals.

Align foundation

Your company's cloud adoption journey is guided with foundational decisions that affect the outcomes of your cloud adoption journey. The following information can help you make core decisions, and record them as a reference to be used during the cloud adoption lifecycle.

- Get started aligning foundation decisions
- How does Azure work
- Fundamental concepts
- Portfolio hierarchy
- Azure hierarchy support

Accelerate adoption

Cloud adoption requires technical change, but to digitally transform with the cloud, it requires more than just IT. Use these guides to start aligning various teams to accelerate migration and innovation efforts.

[] Expand table

Guide	Description
We want to migrate existing workloads to the cloud.	This guide is a great starting point if your primary focus is migrating on-premises workloads to the cloud.
We want to build new products and services in the cloud.	This guide can help you prepare to deploy innovative solutions to the cloud.
We're blocked by environment design and configuration.	This guide provides a quick approach to designing and configuring your environment.

Improve controls

As your cloud adoption journey progresses, a solid operating model can help ensure that wise decisions are made. You'll also want to consider organizational change. These guides help you align people and improve operations to develop your cloud operating model.

[] Expand table

Guide	Description
How do we deliver operational excellence during cloud	The steps in this guide can help the strategy team lead the organizational change management required to consistently

Guide	Description
transformation?	ensure operational excellence.
How do we manage enterprise costs?	This guide can help you start optimizing enterprise costs and manage cost across the environment.
How do we apply the right controls to improve reliability?	This guide helps minimize disruptions related to inconsistencies in configuration, resource organization, security baselines, or resource protection policies.
How do we ensure performance across the enterprise?	This guide can help you establish processes for maintaining performance across the enterprise.

Feedback

Was this page helpful?

 Yes

 No

Get started: Understand and document foundational alignment decisions

Article • 05/06/2024

Cloud adoption provides numerous business, technical, and organizational benefits. Whatever your organization wants to accomplish along its cloud adoption journey, there are certain initial decisions that every team involved in the journey needs to understand. This article provides a list of steps to help you document design decisions and prepare your organization for its cloud adoption process.

ⓘ Note

The links in this article lead to multiple areas of the Cloud Adoption Framework documentation. Bookmarking this article can make it easier for you to find this checklist again after you explore different articles the checklist recommends.

Before you begin

As you work through this guide, use the [initial decision template](#) to record each foundational decision you make. The template helps you clarify the configuration of your cloud environment and the reasons behind each decision. Having this information in one place enables you to rapidly onboard team members participating in your cloud adoption lifecycle.

If you have an environment running in Azure, you can use [Azure Governance Visualizer](#) to accelerate your documentation. The visualizer provides insight into Azure role-based access control (RBAC), infrastructure as code (IaC) such as Terraform or Bicep, policies, and subscriptions. The visualizer also uses collected data to provide visibility into your hierarchy map, create a tenant summary, and build granular scope insights for your management groups and subscriptions.

Step 1: Understand how Azure works

When using Azure as the cloud provider for your cloud adoption journey, you need to understand [how Azure works](#).

Involved teams, deliverables, and supporting guidance:

Everyone involved in your organization's cloud adoption lifecycle should understand what Azure is and how it works.

Step 2: Understand initial Azure concepts

Azure is built on a set of [foundational concepts](#). It would help if you understood these concepts to have in-depth discussions about technical strategy for your Azure implementation.

Involved teams, deliverables, and supporting guidance:

Everyone involved in implementing your organization's Azure technology strategy should understand the terms and definitions of Azure's foundational concepts.

Step 3: Review the portfolio

All cloud hosting and environment decisions require you to understand the portfolio of workloads. Microsoft's Cloud Adoption Framework includes tools to help you understand and evaluate the portfolio.

Deliverables:

- In your [initial decision template](#), record the location and status of the portfolio documentation and who is responsible for managing it.

Guidance to support deliverable completion:

- [Fundamental concepts](#) help you understand critical Azure topics before embarking on your cloud adoption journey.
- The [operations management workbook](#) and business alignment approach help you understand the workloads and assets that transition to your cloud operations team.
- The [cloud adoption plan](#) provides a backlog of workloads and assets slated for cloud adoption.
- The [digital estate analysis](#) approach helps you document existing workloads and assets slated for cloud adoption. In Azure, the digital estate is best represented in the [Azure Migrate](#) tool.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> Your cloud strategy team is accountable for defining a way to view the portfolio. 	<ul style="list-style-type: none"> Multiple teams will use the following guidance to create views. Everyone involved in your organization's cloud adoption should know where to find the portfolio view to support decisions further into the adoption process.

Step 4: Define portfolio-hierarchy depth to align the portfolio

Some organizations can use a single workload and its supporting assets to host their assets and workloads in the cloud. Other organizations might need to include thousands of workloads and many supporting assets for their cloud adoption strategy. The portfolio hierarchy provides common names for each level, creating a common language regardless of which cloud provider an organization decides to use.

Deliverables:

- In your [initial decision template](#), record all relevant hierarchy needs.

Guidance to support deliverable completion:

- Understand the levels of the [portfolio hierarchy](#) so you can align fundamental terms.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> Your cloud governance team is accountable for defining, enforcing, and automating the portfolio hierarchy to shape corporate policy in the cloud. 	<ul style="list-style-type: none"> Everyone involved in your technology strategy for cloud adoption should be familiar with the portfolio hierarchy and the hierarchy levels in use today.

Step 5: Establish naming and tagging standards across the portfolio

All existing workloads and assets should be suitably named and tagged following specific naming and tagging standards. Document these standards and make them

available as a reference for all team members. You should enforce the standards automatically whenever possible to ensure minimum tagging requirements.

Deliverables:

- In your [initial decision template](#), record the location and status of your naming and tagging conventions workbook and who is responsible for managing it.

Guidance to support deliverable completion:

- Create a [naming and tagging standard](#).
- Review and update existing tags in Azure.
- Enforce tagging policies in Azure.

[+] Expand table

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Your cloud governance team is accountable for defining, enforcing, and automating your naming and tagging standards to ensure consistency across the portfolio.	<ul style="list-style-type: none">• Everyone involved in your technology strategy for cloud adoption should be familiar with your naming and tagging standards before cloud deployment.

Step 6: Create a resource organization design to implement the portfolio hierarchy

You must create a resource organization design to ensure consistent alignment with the portfolio hierarchy decisions. This design aligns organizational tools from your cloud provider with the portfolio hierarchy that supports your cloud adoption plan. It also helps guide your implementation by clarifying which assets you can deploy into specific boundaries within a cloud environment.

Deliverables:

- In your [initial decision template](#), map Azure products to the aligned level of the portfolio hierarchy.

Guidance to support deliverable completion:

- Understand how [Azure products support the portfolio hierarchy](#).

- Review your existing subscriptions for alignment with your chosen portfolio hierarchy.

Build a subscription strategy:

- [Start with two subscriptions](#). Add basic subscription designs to account for common enterprise needs like shared services or [sandbox subscriptions](#).
- Ensure you can [manage multiple subscriptions](#) as more subscriptions are needed to support your cloud adoption plan.
- Establish [clear boundaries based on the portfolio hierarchy](#).
- [Move resource groups and assets between subscriptions](#) when necessary to adhere to your organization strategy.

[] [Expand table](#)

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> Your cloud governance team is accountable for defining, implementing, and automating your resource organization design across the portfolio. 	<ul style="list-style-type: none"> Everyone involved in your technology strategy for cloud adoption should be familiar with your resource organization design before cloud deployment.

Step 7: Map capabilities, teams, and RACI to fundamental concepts

Portfolio hierarchy complexity informs organizational structures and methodologies that guide the day-to-day activities of your various teams.

Deliverables:

- Complete the getting started guides for organizational alignment.

Guidance to support deliverable completion:

- Use the previous steps to evaluate the [portfolio hierarchy accountability guidance](#). Determine which capabilities, if any, might need to be delivered by dedicated organizations or virtual teams.
- Apply the portfolio hierarchy accountability guidance to the RACI (responsible, accountable, consulted, and informed) diagram using [Get started: Align your](#)

organization.

[\[+\] Expand table](#)

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">Your cloud strategy team is accountable for aligning virtual or dedicated organizational structures to ensure the success of your cloud adoption lifecycle.	<ul style="list-style-type: none">Everyone involved in your cloud adoption lifecycle should be familiar with the alignment of people and levels of accountability.

Next steps

Follow the guides in the "Get started" section of the Microsoft Cloud Adoption Framework to build on these foundational concepts.

[Apply fundamental concepts to other getting started guides](#)

Feedback

Was this page helpful?

 Yes

 No

How does Azure work?

Article • 11/02/2023

Azure is Microsoft's public cloud platform. Azure offers a large collection of services, which includes platform as a service (PaaS), infrastructure as a service (IaaS), and managed database service capabilities. However, what exactly is Azure, and how does it work?

<https://www.microsoft.com/en-us/videoplayer/embed/RE2ixGo?postJs||Msg=true>

Azure, like other cloud platforms, relies on a technology known as *virtualization*. Most computer hardware can be emulated in software. Computer hardware is simply a set of instructions, which are permanently, or semi-permanently, encoded in silicon. Emulation layers are used to map software instructions to hardware instructions. Emulation layers allow virtualized hardware to execute in software like the actual hardware itself.

Essentially, the cloud is a set of physical servers in one or more datacenters. The datacenters execute virtualized hardware for customers. So how does the cloud create, start, stop, and delete millions of instances of virtualized hardware for millions of customers simultaneously?

To understand the servers, let's look at the architecture of hardware in the datacenter. Inside each datacenter, there's a collection of servers sitting in server racks. Each server rack contains many server blades, and a network switch. These provide network connectivity and a power distribution unit (PDU), which creates power. Racks are sometimes grouped together in larger units known as clusters.

The server racks, or clusters, are chosen to run virtualized hardware instances for the user. However, some servers run cloud management software, known as a fabric controller. The fabric controller is a distributed application with many responsibilities. It allocates services, monitors the health of the server and the services running on it, and heals servers when they fail.

Each instance of the fabric controller is connected to another set of servers running cloud orchestration software, typically known as the front end. The front end hosts the web services, RESTful APIs, and internal Azure databases, which are used for all functions in the cloud.

For example, the front end hosts the services that handle customer requests. The requests allocate Azure resources and services such as [Azure Virtual Machines](#), and [Azure Cosmos DB](#). First, the front end validates and verifies if the user is authorized to allocate the requested resources. If so, the front end checks a database to locate a

server rack with sufficient capacity, which instructs the fabric controller to allocate the resource.

Azure is a huge collection of servers and networking hardware, which runs a complex set of distributed applications. These applications orchestrate the configuration and operation of virtualized hardware and software on those servers. The orchestration of these servers is what makes Azure so powerful. With Azure, users don't have to maintain and upgrade their hardware as Azure does this behind the scenes.

Next steps

Learn about how resources are deployed in Azure with the Azure Resource Manager.

[Microsoft Cloud Adoption Framework for Azure](#)

Azure fundamental concepts

Article • 08/05/2024

In this article, learn about fundamental concepts, terms used in Microsoft Azure, and how the concepts relate to one another.

Azure terminology

It's helpful to know the following definitions as you begin your Azure cloud adoption journey:

- **Resource:** An entity that's managed by Azure. Examples include Azure Virtual Machines, virtual networks, and storage accounts.
- **Subscription:** A logical container for your resources. Each Azure resource is associated with only one subscription. Creating a subscription is the first step in adopting Azure.
- **Azure account:** The email address that you provide when you create an Azure subscription is the Azure account for the subscription. The party that's associated with the email account is responsible for the monthly costs incurred by the resources in the subscription. When you create an Azure account, you provide contact information and billing details, like a credit card. You can use the same Azure account for multiple subscriptions. Each subscription is associated with only one Azure account.
- **Account administrator:** The party associated with the email address that's used to create an Azure subscription. The account administrator is responsible for paying for all costs that incur by the subscription's resources.
- **Microsoft Entra ID:** The Microsoft cloud-based identity and access management service. Microsoft Entra ID lets your employees sign in and access resources.
- **Microsoft Entra tenant:** A dedicated and trusted instance of Microsoft Entra ID. When your organization signs up for a Microsoft cloud service subscription, it automatically creates a Microsoft Entra tenant. For example, Microsoft Azure, Intune, or Microsoft 365. An Azure tenant represents a single organization.
- **Microsoft Entra directory:** Each Microsoft Entra tenant has a single, dedicated, and trusted directory. The directory includes the tenant's users, groups, and applications. Use the directory to manage identity and access management functions for tenant resources. You can associate a directory with multiple subscriptions but each subscription is associated with only one directory.
- **Resource groups:** Logical containers that you use to group related resources in a subscription. Each resource can exist in only one resource group. Resource groups allow for more granular grouping within a subscription. They're commonly used to

represent a collection of assets that are required to support a workload, application, or specific function within a subscription.

- **Management groups:** Logical containers that you use for one or more subscriptions. You can define a hierarchy of management groups, subscriptions, resource groups, and resources to efficiently manage access, policies, and compliance through inheritance.
- **Region:** A set of Azure datacenters that deploy inside a latency-defined perimeter. The datacenters connect through a dedicated, regional, low-latency network. Most Azure resources run in a specific Azure region.

Azure subscription purposes

An Azure subscription serves several purposes, such as:

- **A legal agreement.** Each subscription is associated with an [Azure offer](#), like a free trial or pay-as-you-go. Each offer provides a specific rate plan, benefits, and associated terms and conditions. Choose an Azure offer when you create a subscription.
- **A payment agreement.** When you create a subscription, you provide payment information for that subscription, such as a credit card number. Each month, the costs that incur by the resources deployed to the subscription are calculated and billed to that payment method.
- **A boundary of scale.** Scale limits you define for a subscription. The subscription's resources can't exceed the set scale limits. For example, there's a limit on the number of virtual machines that you can create in a single subscription.
- **An administrative boundary.** A subscription can act as a boundary for administration, security, and policy. Azure also provides other mechanisms to meet these needs, such as management groups, resource groups, and Azure role-based access control.

Azure subscription considerations

When you create an Azure subscription, you make several key choices about the subscription:

- **Who is responsible for paying for the subscription?** By default, the account administrator is the person associated with the email address that you provide when you create a subscription. This person is responsible for paying for all costs incurred by the subscription's resources.
- **Which Azure offer am I interested in?** Each subscription is associated with a specific [Azure offer](#). You can choose the Azure offer that best meets your needs.

For example, if you intend to use a subscription to run non-production workloads, you might choose the Pay-As-You-Go Dev/Test offer, or the Enterprise Dev/Test offer.

Note

When you sign up for Azure, you might see the phrase *Create an Azure account*. You create an Azure account when you create an Azure subscription. You can associate the subscription with an email account.

Azure administrative roles

Azure defines three types of roles for administering subscriptions, identities, and resources:

- Classic subscription administrator roles
- Azure roles
- Microsoft Entra roles

The account administrator role is assigned to the email account that's used to create the Azure subscription. The account administrator is the billing owner of the subscription. The account administrator can [manage subscription administrators](#) in the Azure portal.

By default, the service administrator role for a subscription is also assigned to the email account that's used to create the Azure subscription. The service administrator has permissions to the subscription equivalent to the Azure role-based access control Owner role. The service administrator also has full access to the Azure portal. The account administrator can change the service administrator to a different email account.

When you create an Azure subscription, you can associate it with an existing Microsoft Entra tenant. You provide an email account and can associate that email account with multiple Azure subscriptions. An account administrator can transfer a subscription to another account. For more information, see [Classic subscription administrator roles, Azure roles, and Microsoft Entra roles](#).

Subscriptions and regions

Every Azure resource is logically associated with one subscription. When you create a resource, you choose which Azure subscription to deploy that resource to. You can move a resource to another subscription later.

Subscriptions aren't tied to a specific Azure region, but each Azure resource deploys to only one region. You can have resources in multiple regions that are associated with the same subscription.

Note

Most Azure resources deploy to a specific region. Certain resource types are considered global resources, such as policies that you set by using the Azure Policy services.

Related resources

The following resources provide detailed information about the concepts discussed in this article:

- [How does Azure work?](#)
- [Resource access management in Azure](#)
- [Azure Resource Manager overview](#)
- [Azure role-based access control \(Azure RBAC\)](#)
- [What is Microsoft Entra ID?](#)
- [Associate or add an Azure subscription to your Microsoft Entra tenant](#)
- [Topologies for Microsoft Entra Connect](#)
- [Subscriptions, licenses, accounts, and tenants for Microsoft's cloud offerings](#)

Next steps

Now that you understand fundamental Azure concepts, learn how to scale with multiple Azure subscriptions.

[Scale with multiple Azure subscriptions](#)

Feedback

Was this page helpful?

 Yes

 No

Resource access management in Azure

Article • 10/06/2023

In this article, learn how resources are deployed in Azure, starting with the fundamental Azure constructs of resources, subscriptions and resource groups. You will then learn how Azure Resource Manager (ARM) deploys resources.

What is an Azure resource?

In Azure, a resource is an entity managed by Azure. Virtual machines, virtual networks, and storage accounts are all examples of Azure resources.



What is an Azure resource group?

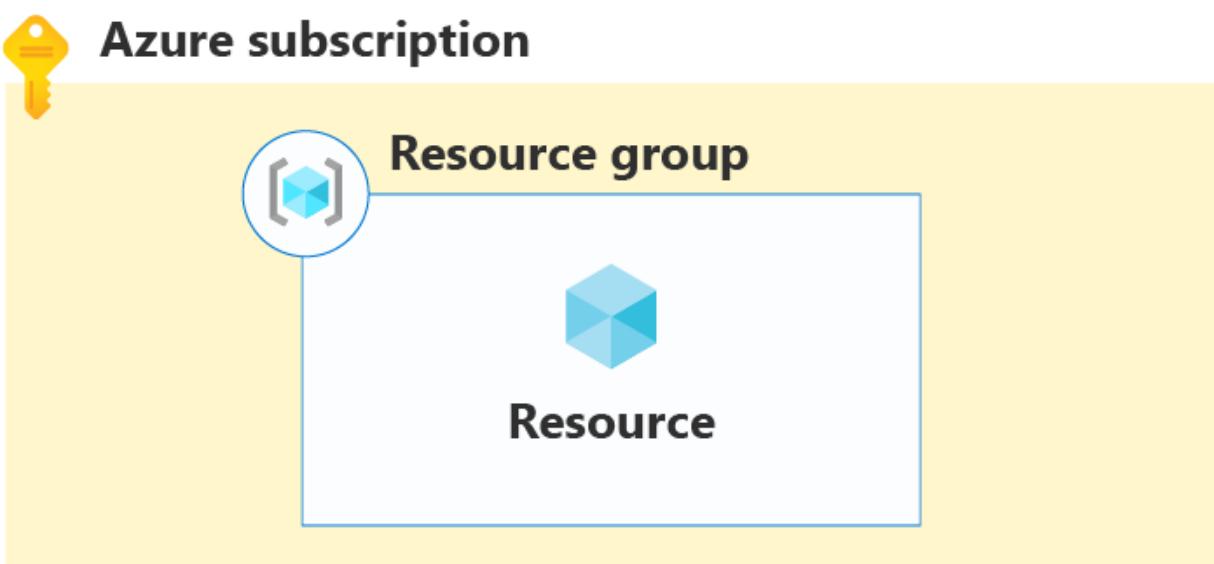
Each resource in Azure must belong to a [resource group](#). A resource group is a logical container that associates multiple resources so you can manage them as a single entity, based on lifecycle and security. For example, you can create or delete resources as a group if the resources share a similar lifecycle, such as the resources for an [n-tier application](#). In other words, everything that you create, manage, and deprecate together is associated within a resource group.



Recommended best practice is to associate resource groups, and the resources they contain, with an Azure subscription.

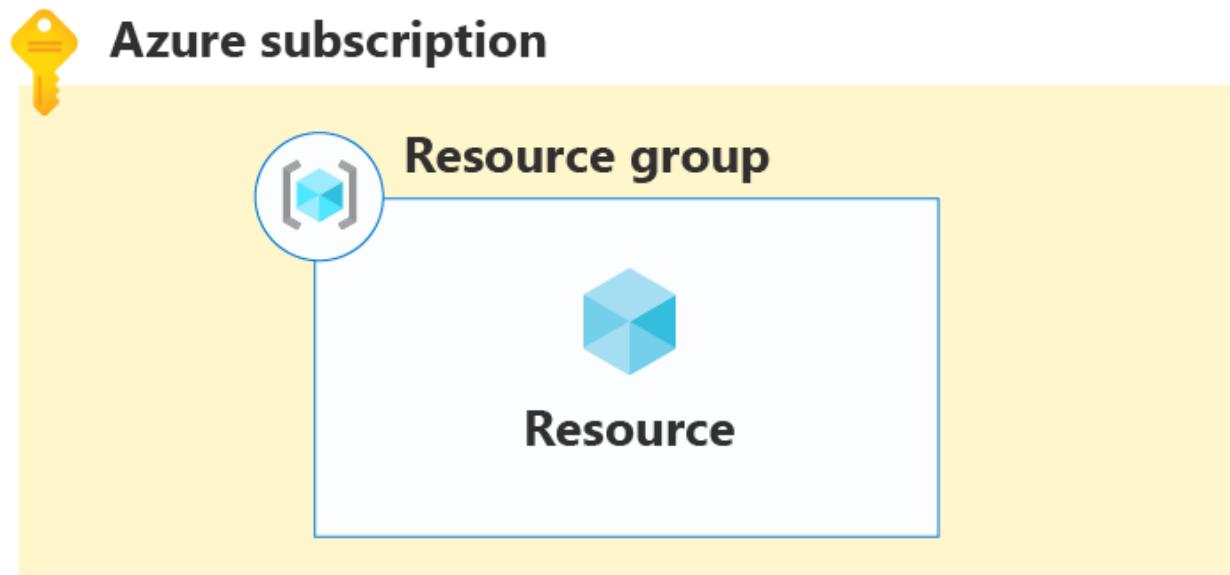
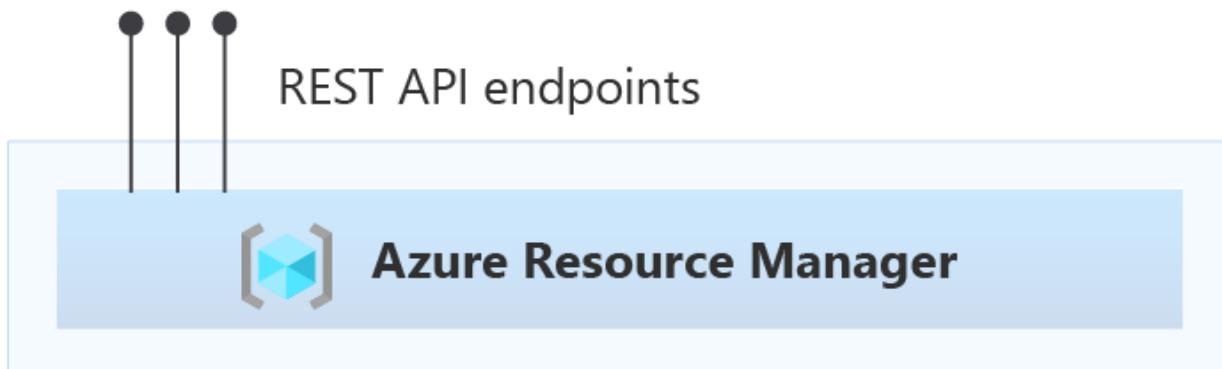
What is an Azure subscription?

An Azure subscription is similar to a resource group in that it's a logical container that associates resource groups and their respective resources. An Azure subscription is also associated with Azure Resource Manager controls. Learn about [Azure Resource Manager](#) and its relationship to Azure subscriptions.



What is Azure Resource Manager?

In [How does Azure work?](#), you learn that Azure includes a front end with services that orchestrate Azure's functions. One of these services is [Azure Resource Manager](#). This service hosts the RESTful API clients use to manage resources.



The following figure shows three clients: [Azure PowerShell](#), the [Azure portal](#), and the [Azure CLI](#):



Azure
PowerShell



Azure
portal



Azure
CLI



REST API endpoints



Azure Resource Manager



Azure subscription



Resource group



Resource

While these clients connect to Resource Manager using the REST API, Resource Manager doesn't include functionality to manage resources directly. Rather, most resource types in Azure have their own [resource provider](#).



Azure
PowerShell



Azure
portal



Azure
CLI



REST API endpoints



Azure Resource Manager

Azure resource providers



• • •



Azure subscription

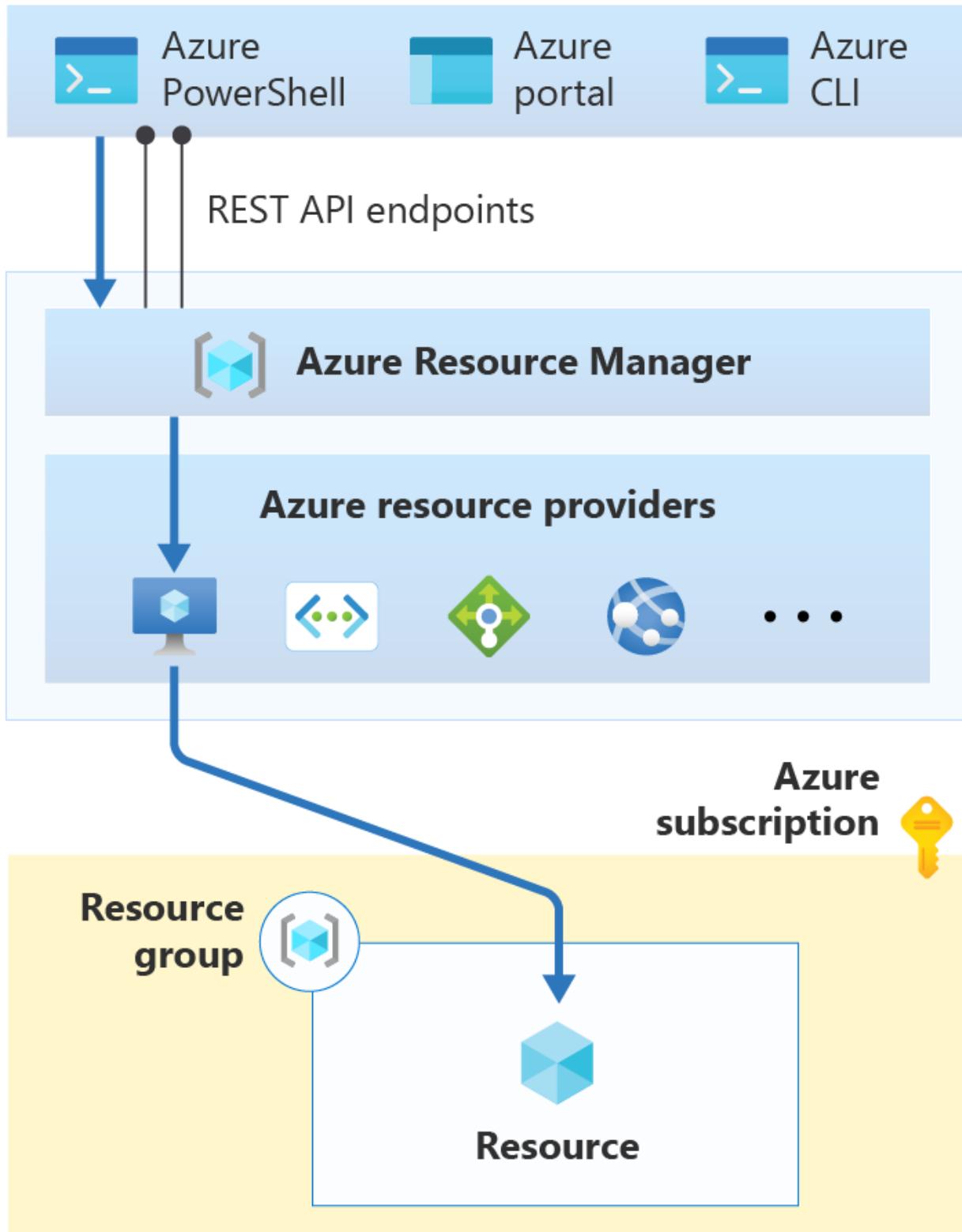


Resource group



Resource

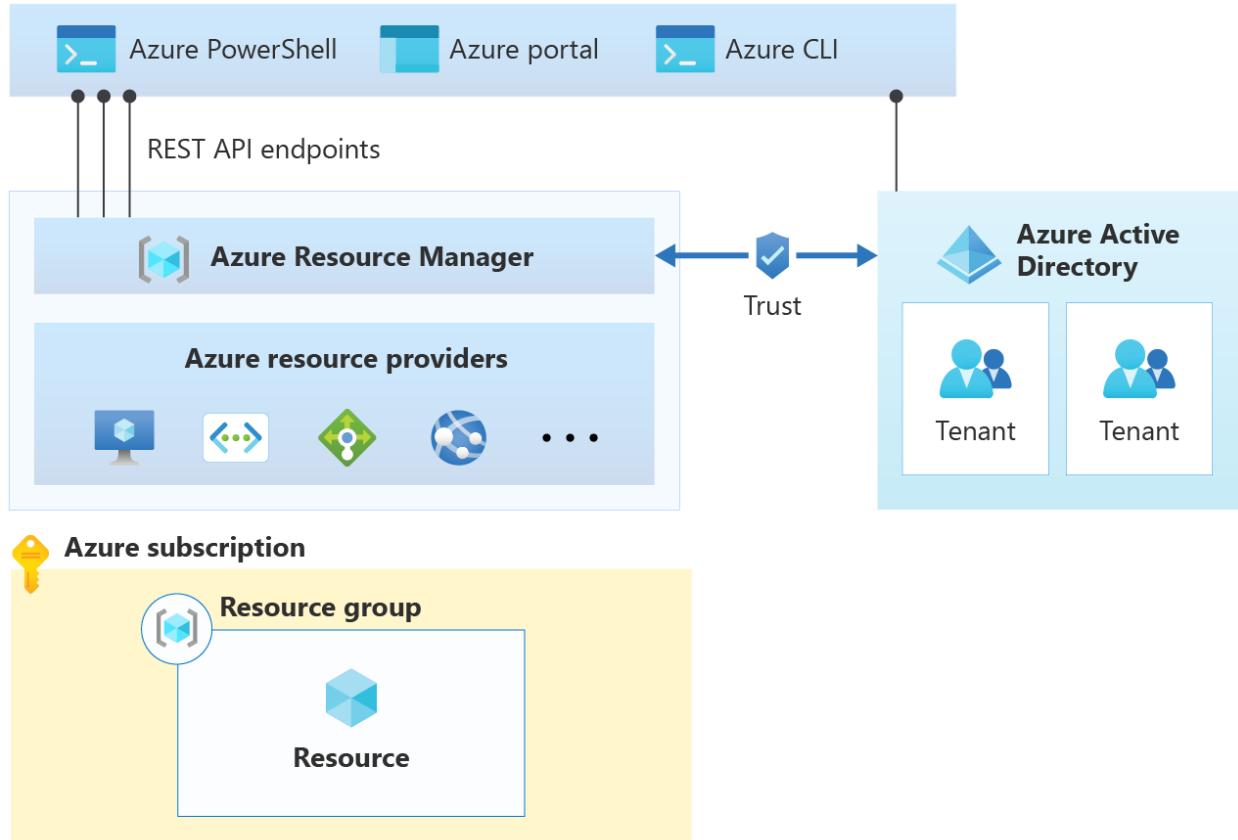
When a client makes a request to manage a specific resource, Azure Resource Manager connects to the resource provider for that resource type to complete the request. For example, if a client makes a request to manage a virtual machine resource, Azure Resource Manager connects to the `Microsoft.Compute` resource provider.



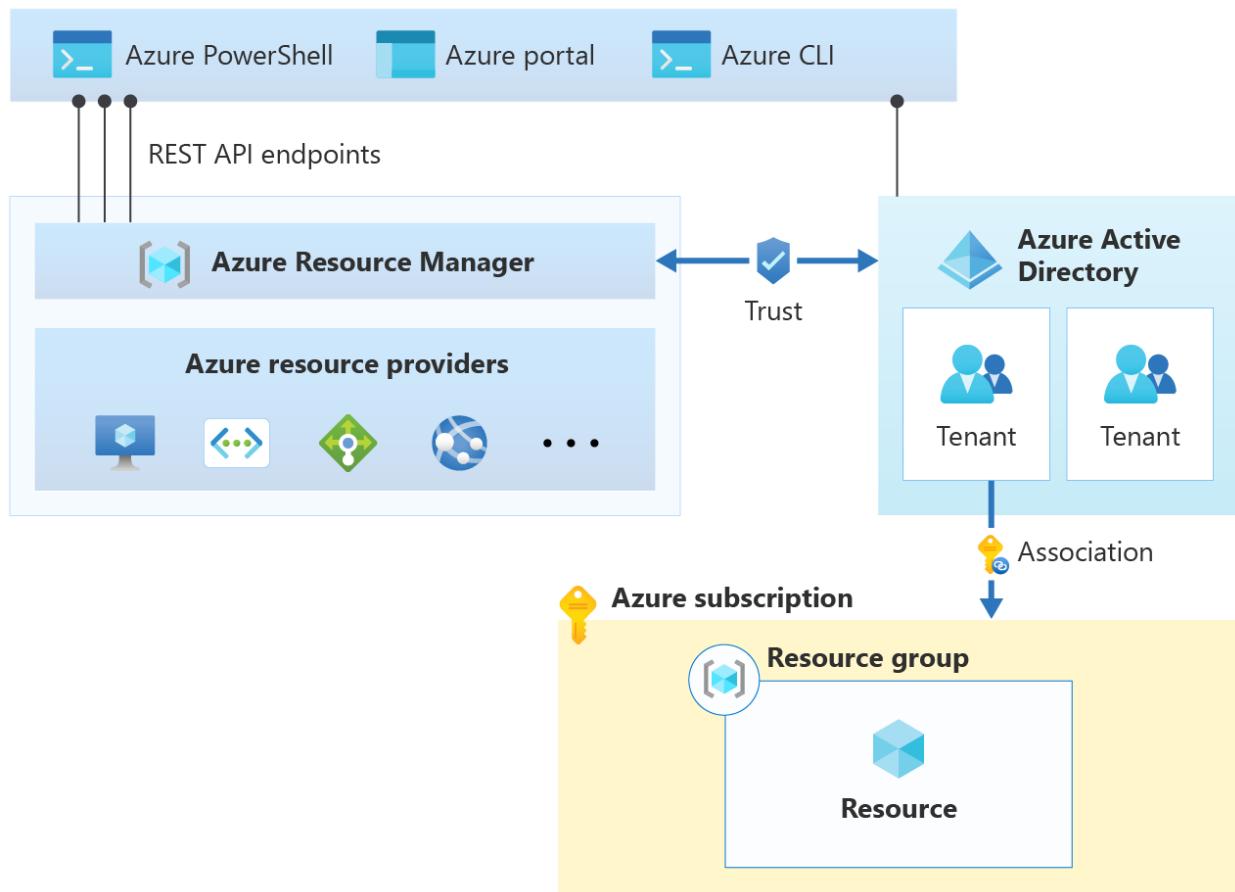
Azure Resource Manager requires the client to specify an identifier for both the subscription and the resource group to manage the virtual machine resource.

Once you understand how Azure Resource Manager works, you can learn how to associate an Azure subscription with the Azure Resource Manager controls. Before Azure Resource Manager can execute any resource management request, review the following a set of controls.

The first control is that a validated user must make a request. Also, Azure Resource Manager must have a trusted relationship with [Microsoft Entra ID](#) to provide user identity functionality.

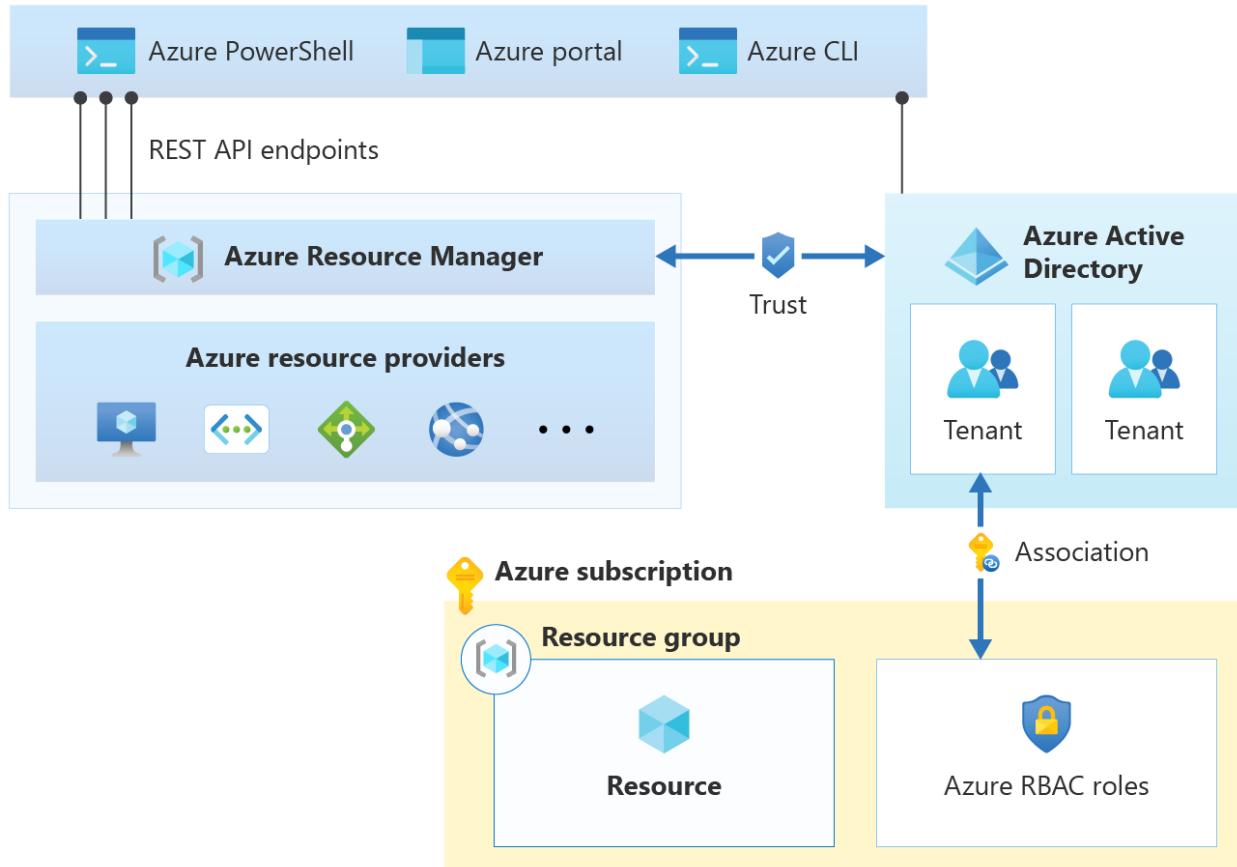


In Microsoft Entra ID, you can segment users into tenants. A *tenant* is a logical construct that represents a secure, dedicated instance of Microsoft Entra ID that someone typically associates with an organization. You can also associate each subscription with a Microsoft Entra tenant.



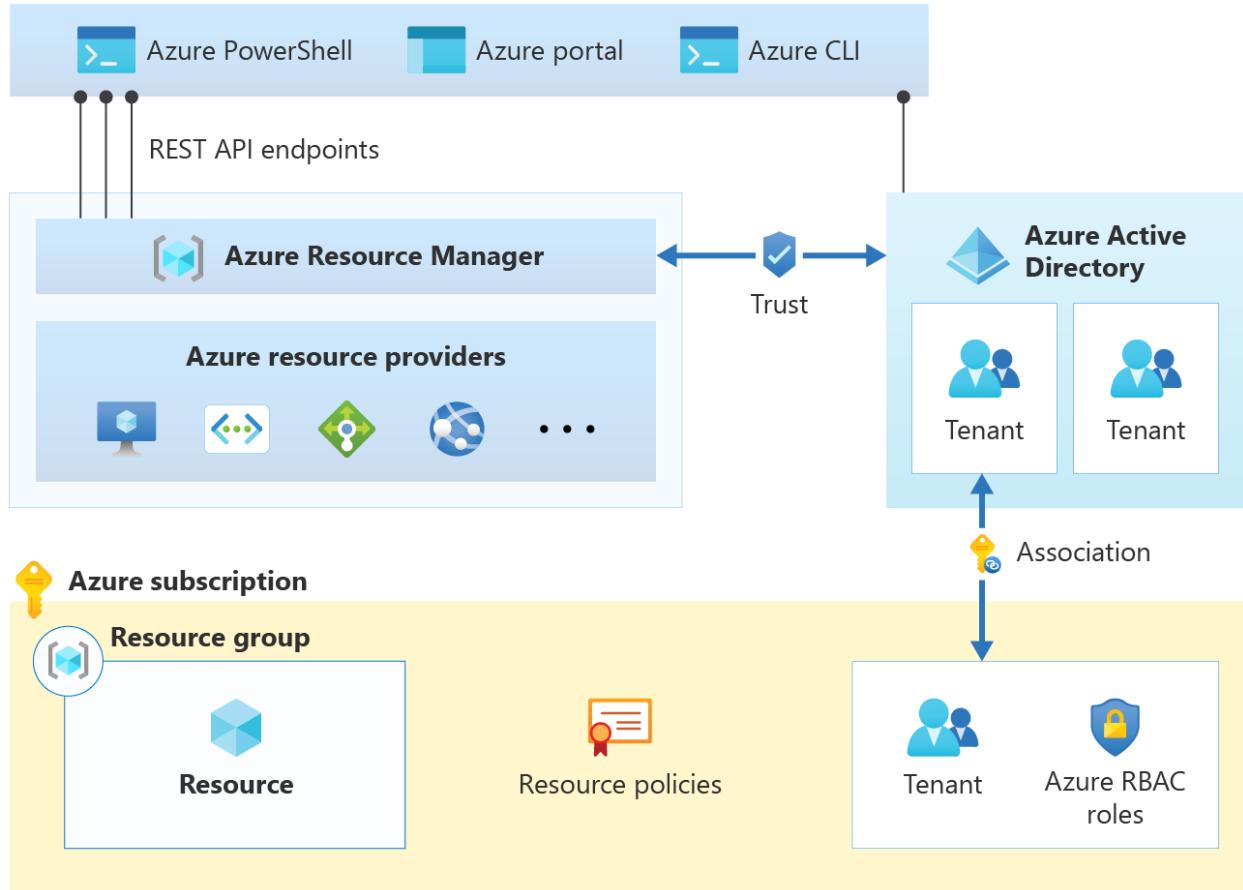
Each client request to manage a resource in a particular subscription requires that the user has an account in the associated Microsoft Entra tenant.

The next control is a check that the user has sufficient permission to make the request. Permissions are assigned to users using [Azure role-based access control \(Azure RBAC\)](#).

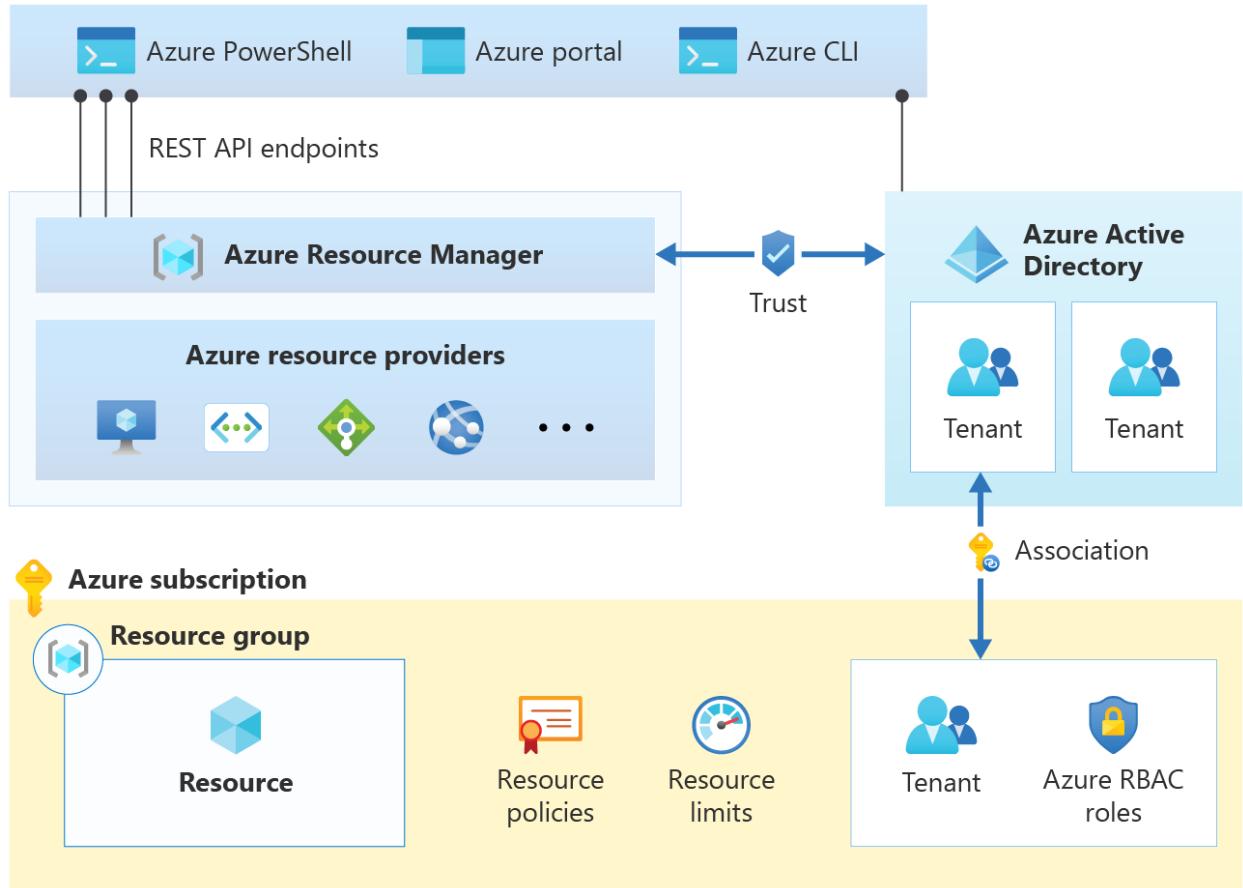


An Azure role specifies a set of permissions a user can take on a specific resource. When the role is assigned to the user, those permissions are applied. For example, the [built-in Owner role](#) allows a user to run any action on a resource.

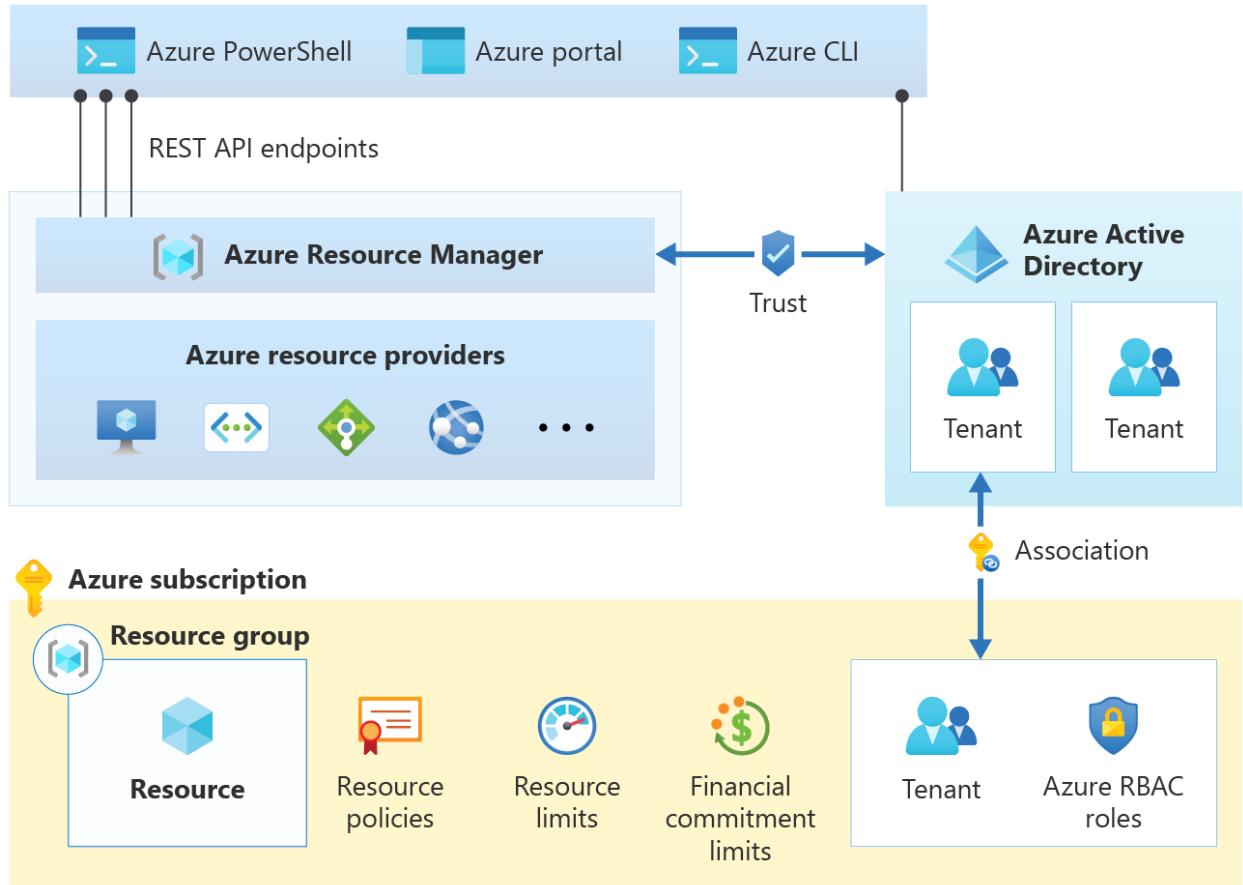
The next control is a check that the request is allowed under the settings specified for [Azure resource policy](#). Azure resource policies specify the operations allowed for a specific resource. For example, an Azure resource policy can specify that users are only allowed to deploy a specific type of virtual machine.



The next control is a check that the request doesn't exceed an [Azure subscription limit](#). For example, each subscription has a limit of 980 resource groups per subscription. If you receive a request to deploy another resource group when the limit has been reached, deny it.



The final control is a check to verify the request is within the financial commitment that you associate with the subscription. For example, Azure Resource Manager verifies the subscription has sufficient payment information if the request is to deploy a virtual machine.



Summary

In this article, you learned about how resource access is managed in Azure using Azure Resource Manager.

Next steps

Learn more about cloud adoption with the Microsoft Cloud Adoption Framework for Azure.

[Microsoft Cloud Adoption Framework for Azure](#)

Portfolio hierarchy

Article • 02/28/2023

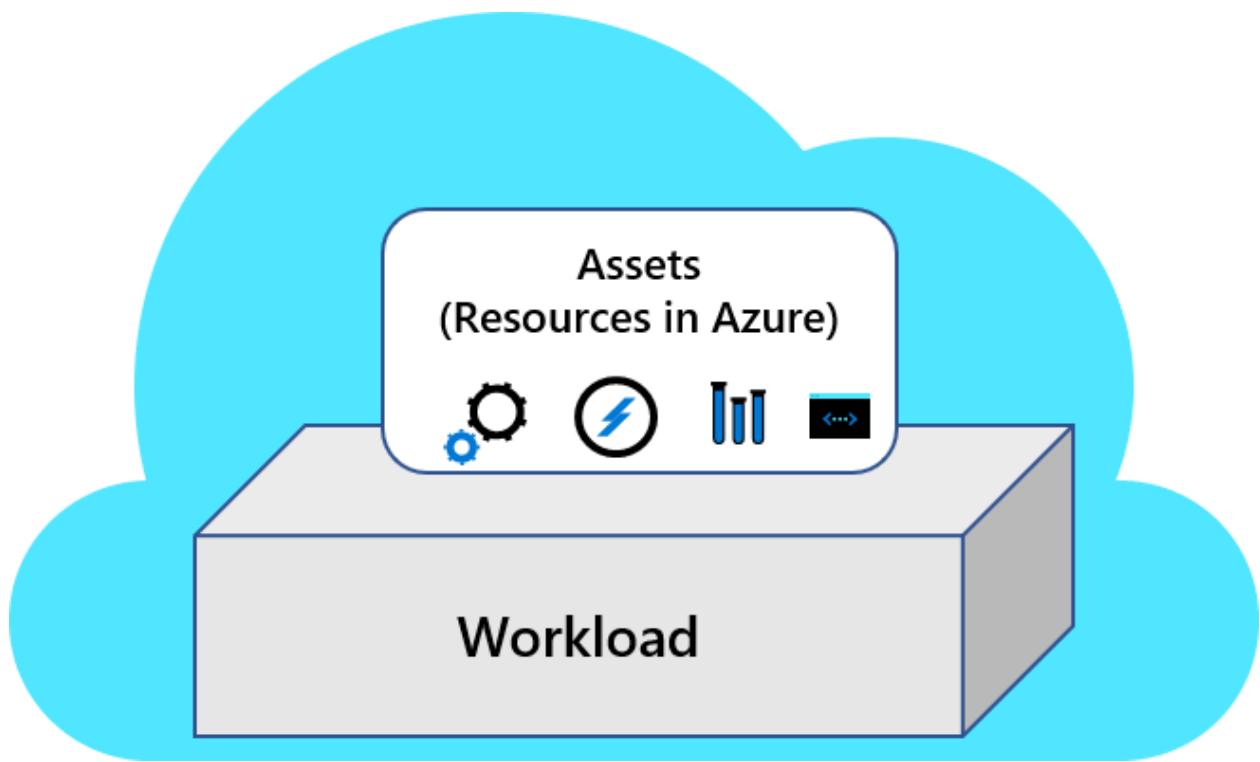
To understand how a *portfolio* of workloads, assets, and supporting services all work together, you need to establish a portfolio hierarchy. This article provides clear definitions for the levels of your portfolio hierarchy, along with guidance for teams to deliver on the expectations for each level. Throughout this article, each level of the hierarchy is also called a *scope*.

Workloads

A collection of technologies that delivers defined business value is called a *workload*. The collection might include applications, servers or virtual machines, data, devices, and other similarly grouped assets. Most businesses rely on multiple workloads to deliver vital business functions.

Typically, a business stakeholder and technical leader share accountability for the ongoing support of each workload. In some phases of the workload lifecycle, those roles are clearly stated. In more operational phases of a workload's lifecycle, those roles might be transitioned to a shared operations management team or cloud operations team. As the number of workloads increases, the roles (stated or implied) become more complex and more matrixed.

Workloads and their supporting assets are at the core of any portfolio. The scopes or layers define how those workloads are viewed and to what extent they're affected by the matrix of potential supporting teams.



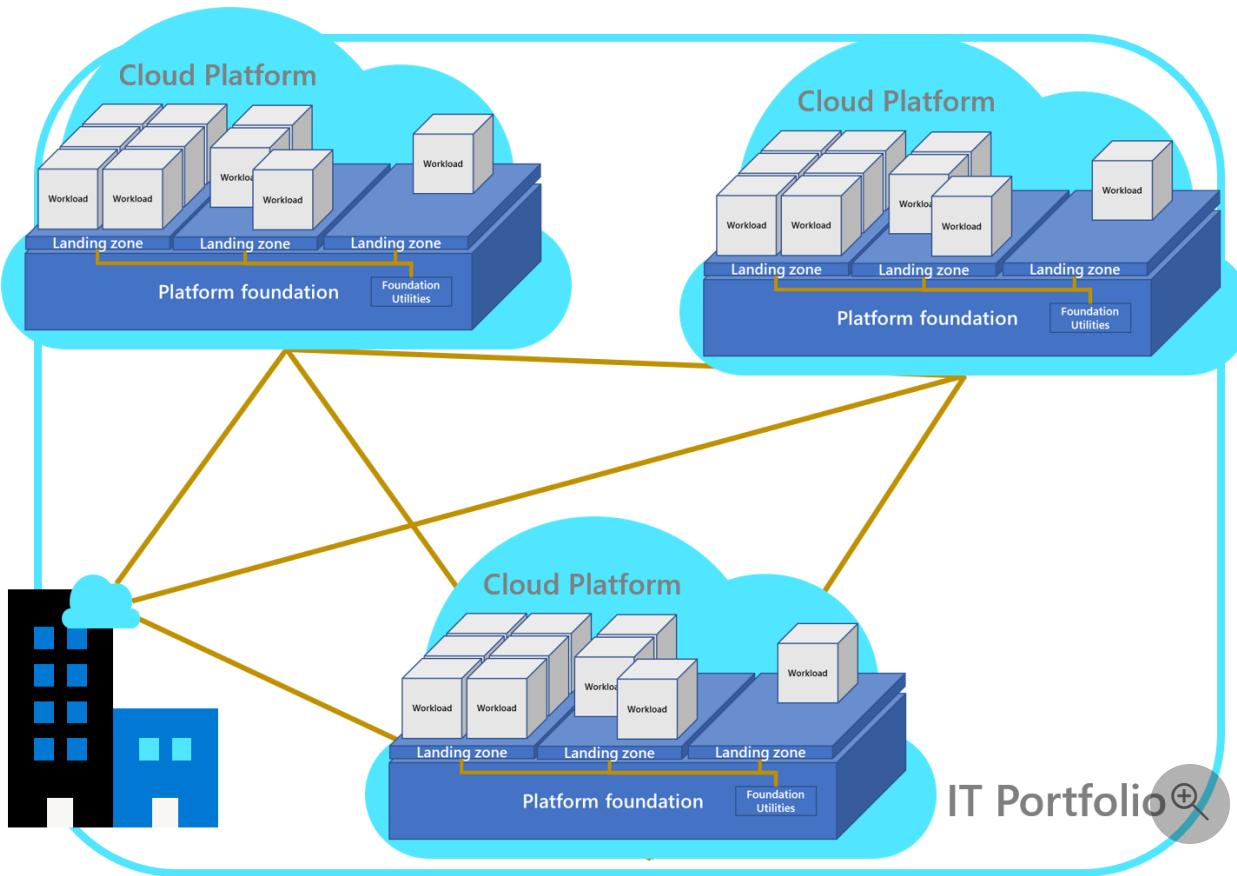
Although the terms can vary, all IT solutions include assets and workloads:

- **Asset:** The smallest unit of technical function that supports a workload or solution.
- **Workload:** The smallest unit of IT support for the business. A workload is a collection of infrastructure, applications, and data assets that support a common business goal or the execution of a common business process.

When you're deploying your first workload, the workload and its assets might be the only defined scope. The other layers might be explicitly defined as more workloads are deployed.

IT portfolio

When companies support workloads through matrixed approaches or centralized approaches, a broader hierarchy likely exists to support those workloads:



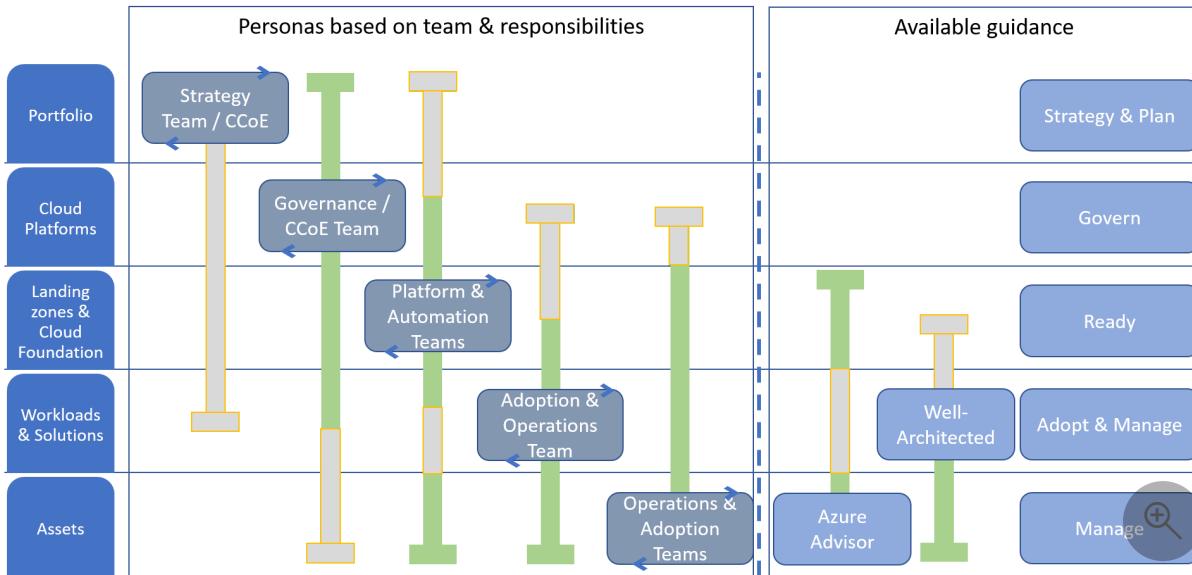
- **Landing zones:** Landing zones provide workloads with the necessary foundational utilities, or shared plumbing, that are required to support one or more workloads. Landing zones are so critical in the cloud that the entire *Ready* methodology of the Cloud Adoption Framework focuses on landing zones. For a more detailed definition, see [What is a landing zone?](#)
- **Foundational utilities:** These shared IT services are required for workloads to operate within the technology and business portfolio.
- **Platform foundation:** This organizational construct centralizes foundational solutions and helps ensure that those controls are enforced for all landing zones.
- **Cloud platforms:** Depending on the overall strategy for supporting the full portfolio, customers might need multiple cloud platforms with distinct deployments of the platform foundation to govern multiple regions, hybrid solutions, or even multicloud solutions.
- **Portfolio:** Through a technology lens, the portfolio is a collection of workloads, assets, and supporting resources that span all cloud platforms. Through a business lens, the portfolio is the collection of projects, people, processes, and investments that support and manage the technology portfolio to drive business outcomes. Together, these two lenses capture the portfolio.

Hierarchy accountability and guidance

An accountable team manages each layer of the portfolio hierarchy. The following diagram shows the mapping between the accountable team and the guidance to support its business decisions, technical decisions, and technical implementation.

ⓘ Note

The teams mentioned in the following list might be virtual teams or individuals. For some variants of this hierarchy, some of the accountable teams can be collapsed as described later in the accountability variants.



- **Portfolio:** The cloud strategy team and the cloud center of excellence (CCoE) use the *Strategy* and *Plan* methodologies to guide decisions that affect the overall portfolio. The cloud strategy team is accountable for the enterprise level of the cloud portfolio hierarchy. The cloud strategy team should also be informed of decisions about the environment, landing zones, and high-priority workloads.
- **Cloud platforms:** The cloud governance team is accountable for the disciplines that ensure consistency across each environment in alignment with the *Govern* methodology. The cloud governance team is accountable for governance of all resources in all environments. The cloud governance team should be consulted on changes that might require an exception or change to governing policies. The cloud governance team should also be informed of progress with workload and asset adoption.
- **Landing zones and cloud foundation:** The cloud platform team is accountable for developing the landing zones and platform utilities that support adoption. The cloud automation team is accountable for automating the development of, and ongoing support for, those landing zones and platform utilities. Both teams use the *Ready* methodology to guide implementation. Both teams should be informed

of progress with workload adoption and any changes to the enterprise or environment.

- **Workloads:** Adoption happens at the workload level. Cloud adoption teams use the *Migrate* and *Innovate* methodologies to establish scalable processes to accelerate adoption. After adoption is complete, the ownership of workloads is likely transferred to a cloud operations team that uses the *Manage* methodology to guide operations management. Both teams should be comfortable using the Microsoft Azure Well-Architected Framework to make detailed architectural decisions that affect the workloads they support. Both teams should be informed of changes to landing zones and environments. Both teams might occasionally contribute to landing zone features.
- **Assets:** Assets are typically the responsibility of the cloud operations team. That team uses the management baseline in the *Manage* methodology to guide operations management decisions. It should also use Azure Advisor and the Azure Well-Architected Framework to make detailed resource and architectural changes that are required to deliver on operations requirements.

Accountability variants

- **Single environment:** When an enterprise needs only one environment, a CCoE is typically not required.
- **Single landing zone:** If an environment has only a single landing zone, the governance and platform capabilities can likely be combined into one team.
- **Single workload:** Some businesses need only one workload, or a few workloads, in a single landing zone and a single environment. In those cases, there's little need for a separation of duties between governance, platform, and operations teams.

Common workload and accountability examples

The following examples illustrate portfolio hierarchy.

COTS workloads

Traditionally, enterprises have favored commercial-off-the-shelf (COTS) software solutions to power business processes. These solutions are installed, configured, and then operated. There's little change to the solutions architecture after configuration.

In these scenarios, any cloud adoption of COTS solutions ends with a transition to a cloud operations team. The cloud operations team then becomes the technical owner

for that software and assumes accountability for managing configuration, cost, patching cycles, and other operational needs.

These workloads include accounting packages, logistics software, or industry-specific solutions. In Microsoft terminology, the vendors of these packages are called independent software vendors (ISVs). Many ISVs offer a service to deploy and maintain an instance of their software package in your subscriptions. They might also offer a version of the software package that runs in their own cloud-hosted environment, providing a platform as a service (PaaS) alternative to the workload.

Except for PaaS offerings, cloud operations teams are responsible for ensuring basic operational compliance requirements for those workloads. A cloud operations team should work with the cloud governance team to align cost, performance, and other architecture pillars.

In development with active revisions

When a COTS solution or PaaS offering isn't aligned to the business need, or no solution exists, enterprises build custom-developed workloads. Typically, only a small percentage of the IT portfolio uses this workload approach. But these workloads tend to drive a disproportionately high percentage of IT's contribution to business outcomes, especially outcomes related to new revenue streams. These workloads tend to map well to new innovation ideas.

Given various movements that are rooted in agile methodologies and DevOps practices, these workloads favor a business/DevOps alignment over traditional IT management. For these workloads, there might not be a handoff to the cloud operations team for several years. In those cases, the development team serves as the technical owner of the workload.

Due to extensive time and capital constraints, custom development options are often limited to high-value opportunities. Typical examples include application innovations, deep data analysis, or mission-critical business functions.

Break/fix or sunset development

After a custom-developed workload reaches peak maturity, the development team might be reassigned to other projects. In these cases, technical ownership typically transitions to a cloud operations team. When there's a need for small fixes, the operations team will enlist developers to resolve the error.

In some cases, the development team moves to a project that will eventually replace the current workload. Alternatively, the team might move on because the business opportunity supported by the workload is being phased out. In these sunset scenarios, the cloud operations team serves as the technical owner until the workload is no longer needed.

In both scenarios, the cloud operations team typically serves as the long-term technical owner and decision maker. That team will likely enlist application developers when operational changes require significant architectural changes.

Mission-critical workloads

In every company, a few workloads are too important to the business for them to fail. With these mission-critical workloads, there are usually operations and development owners with various levels of responsibility. Those teams should align operational changes and architectural changes to minimize disruptions to the production solution.

These scenarios require a strong focus on separation of duties. The operations team will generally hold accountability for day-to-day operational changes in the production environment. When those operational changes require an architectural change, they'll be completed by the development or adoption team in a nonproduction environment, before the operations team applies the changes to production.

Examples of mission-critical workloads with a required separation of duties include workloads like SAP, Oracle, or other enterprise resource planning (ERP) solutions, which span multiple business units in the company.

Strategy portfolio alignment

It's important to understand the strategic objectives of the cloud adoption effort and align the portfolio to support that transformation. A few common types of strategic portfolio alignment help shape the structure of the portfolio hierarchy. The following sections provide examples of the portfolio alignment and effect on the portfolio hierarchy.

Innovation or development-led portfolio

Some companies, especially fast-growing startups, have a higher-than-average percentage of custom development projects. In development-heavy portfolios, the environment, landing zone, and workloads are often compressed, so there might be

specific environments for specific workloads. The result is a 1:1 ratio between environment, landing zone, and workload.

Because the environment hosts custom solutions, the DevOps pipeline and application-level reporting might replace the need for operations and governance functions. Those customers will likely reduce focus on operations, governance, or other supporting roles. A stronger emphasis on the responsibilities of the cloud adoption and cloud automation teams is also typical.

Portfolio alignment: The IT portfolio will likely focus on workloads and workload owners to drive critical architecture decisions. Those teams are likely to find more value in the Azure Well-Architected Framework guidance during adoption and operations activities.

Boundary definitions: The logical boundaries, even at an enterprise level, will likely focus on production and nonproduction environment segmentation. There might also be clear segmentation between products in the company's software portfolio. At times, there might also be segmentation between development and hosted customer instances.

Operations-led portfolio

Multinational enterprise organizations with more established IT operations teams typically have a stronger focus on governance and operations than development. In these organizations, a higher percentage of workloads typically aligns to the COTS or break/fix categories, maintained by technical owners within the cloud operations team.

Portfolio alignment: The IT portfolio will be workload aligned, but those workloads are then aligned to operating units or business units. There might also be organization around funding models, industry, or other business segmentation requirements.

Boundary definitions: Landing zones will likely group applications into application archetypes to keep similar operations in a similar segmentation. Environments will likely refer to physical constructs like datacenter, nation, cloud-provider region, or other operational organization standards.

Migration-led portfolio

Similar to operations-led portfolios, a portfolio that is largely built through migration will be based on specific business drivers that led to the migration of existing assets. Typically, the datacenter is the biggest factor in those drivers.

Portfolio alignment: The IT portfolio might be workload aligned, but it's more likely asset aligned. If transitions to IT operations have happened in the company's history,

many active-use assets might not be easily mapped to a funded workload. In these cases, many assets might not have a defined workload or clear workload owner until late in the migration process.

Boundary definitions: Landing zones will likely group applications into boundaries that reflect on-premises segmentation. Though not a best practice, environments often match the on-premises datacenter name and landing zones that represent network segmentation practices. It's a better practice to adhere to segmentation that more closely aligns with an operations-led portfolio.

Governance-led portfolio

Alignment to governance teams should happen as early as possible. Through governance practices and cloud governance tooling, portfolios and environmental boundaries can best balance the needs of innovation, operations, and migration efforts.

Portfolio alignment: Governance-led portfolios tend to include data points that capture innovation and operations details, such as workload, operations owner, data classification, and operational criticality. These data points create a well-rounded view of the portfolio.

Boundary definitions: Boundaries in a governance-led portfolio tend to favor operations over innovation, while using a management-group-driven hierarchy that maps to criteria for business units and development environments. At each level of the hierarchy, a cloud governance boundary can have different degrees of policy enforcement to allow for development and creative flexibility. At the same time, production-grade requirements can be applied to production subscriptions to ensure separation of duties and consistent operations.

Next steps

Learn how [Azure products support the portfolio hierarchy](#).

How do Azure products support the portfolio hierarchy?

Article • 02/28/2023

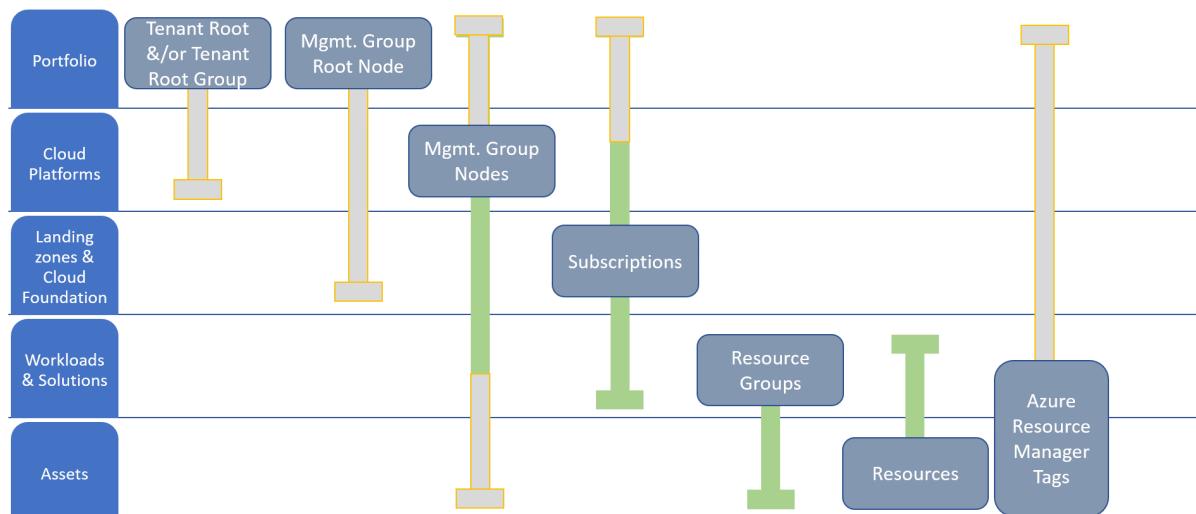
In [Understanding and aligning the portfolio hierarchy](#), a set of definitions for the portfolio hierarchy and role mapping established a hierarchy of scope for most portfolio approaches. As described in that article, you might not need each of the outlined levels or *scopes*. Minimizing the number of layers reduces complexity, so these layers shouldn't all be viewed as a requirement.

This article shows how each level or scope of the hierarchy is supported in Azure through organizational tools, deployment and governance tools, and some solutions in the Microsoft Cloud Adoption Framework for Azure.

Organizing the hierarchy in Azure

Azure Resource Manager includes several organizational approaches that help organize assets at each level of the cloud hierarchy.

The slide bars in the following diagram demonstrate common variants in alignment. The gray parts of the slide bars are common but should be used only for specific business requirements. The points after the image describe a suggested best practice.



- **Portfolio:** The enterprise or business unit probably won't contain any technical assets but might affect cost decisions. The enterprise and business units are represented in the root nodes of the management group hierarchy.

- **Cloud platforms:** Each environment has its own node in the management group hierarchy.
- **Landing zones and cloud foundation:** Each landing zone is represented as a subscription. Likewise, platform foundations are contained in their own subscriptions. Some subscription designs might call for a subscription per cloud or per workload, which would change the organizing tool for each.
- **Workloads:** Each workload is represented as a resource group. Resource groups are often used to represent solutions, deployments, or other technical groupings of assets.
- **Assets:** Each asset is inherently represented as a resource in Azure.

Organizing with tags

Deviations from the best practice are common. You can record them by tagging all assets. Use a tag to represent each of the relevant layers of the hierarchy. For more information, see [Recommended naming and tagging conventions](#).

Get started: Accelerate migration

Article • 04/03/2023

Proper alignment of business and IT stakeholders helps overcome migration roadblocks and accelerate migration efforts. This article provides recommended steps for:

- Stakeholder alignment
- Migration planning
- Deploying a landing zone
- Migrating your first 10 workloads

This article also helps you implement proper governance and management processes. Use this guide to streamline the processes and materials required to align an overall migration effort.

If your migration scenario is atypical, you can get a personalized assessment of your organization's migration readiness by using the [strategic migration and readiness tool \(SMART\) assessment](#). Use it to identify the guidance that best aligns to your current needs.

Get started

The technical effort and process required to migrate workloads is relatively straightforward. It's important to complete the migration process efficiently. Strategic migration readiness has an even bigger impact on the timelines and successful completion of the overall migration.

To accelerate adoption, you must take steps to support the cloud adoption team during migration. This guide outlines these iterative tasks to help customers start on the right path toward any cloud migration. To show the importance of the supporting steps, migration is listed as step 10 in this article. In practice, the cloud adoption team is likely to begin their first pilot migration in parallel with steps 4 or 5.

Step 1: Align stakeholders

To avoid common migration blockers, create a clear and concise business strategy for migration. Stakeholder alignment on motivations and expected business outcomes shapes decisions made by the cloud adoption team.

- **Motivations:** The first step to strategic alignment is to gain agreement on the motivations that drive the migration effort. Start by understanding and

categorizing motivations and common themes from various stakeholders across business and IT.

- **Business outcomes:** After motivations are aligned, it's possible to capture the desired business outcomes. This information provides clear metrics you can use to measure the overall transformation.

Deliverables:

- Use the [strategy and plan template](#) ↗ to record motivations and desired business outcomes.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud strategy team	<ul style="list-style-type: none">• Cloud adoption team• Cloud center of excellence or central IT team

Step 2: Align partner support

Partners, Microsoft Services, or various Microsoft programs are available to support you throughout the migration process.

- [Understand partnership options](#) to find the right level of partnership and support.

Deliverables:

- Establish terms and conditions or other contractual agreements before you engage supporting partners.
- Identify approved partners in the [strategy and plan template](#) ↗ .

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud strategy team	<ul style="list-style-type: none">• Cloud adoption team• Cloud center of excellence or central IT team

Step 3: Gather data and analyze assets and workloads

Use discovery and assessment to improve technical alignment and create an action plan for executing your strategy. During this step, validate the business case using data about

the current state environment. Then perform quantitative analysis and a deep qualitative assessment of the highest priority workloads.

- **Inventory existing systems:** Use a programmatic data-driven approach to understand the current state. Discover and gather data to enable all assessment activities.
- **Incremental rationalization:** Streamline assessment efforts to focus on a qualitative analysis of all assets, possibly even to support the business case. Then add a deep qualitative analysis for the first 10 workloads to be migrated.

Deliverables:

- Raw data on existing inventory.
- Quantitative analysis on existing inventory to refine the business justification.
- Qualitative analysis of the first 10 workloads.
- Business justification documented in the [strategy and plan template](#).

Accountable team	Responsible and supporting teams
• Cloud adoption team	• Cloud strategy team

Step 4: Make a business case

Making the business case for migration is likely to be an iterative conversation among stakeholders. In this first pass at building the business case, evaluate the initial high-level return from a potential cloud migration. The goal of this step is to ensure that all stakeholders align around one simple question: based on the available data, is the overall adoption of the cloud a wise business decision?

- **Building a cloud migration business case** is a good starting point for developing a migration business case. Clarity on formulas and tools can aid in business justification.

Deliverables:

- Use the [strategy and plan template](#) to record business justification.

Accountable team	Responsible and supporting teams
• Cloud strategy team	• Cloud adoption team

Step 5: Create a migration plan

A cloud adoption plan provides an accelerated approach to developing a project backlog. The backlog can then be modified to reflect discovery results, rationalization, needed skills, and partner contracting.

- [Cloud adoption plan](#): Define your cloud adoption plan using the basic template.
- [Workload alignment](#): Define workloads in the backlog.
- [Effort alignment](#): Align assets and workloads in the backlog to clearly define effort for prioritized workloads.
- [People and time alignment](#): Establish iteration, velocity (people's time), and releases for the migrated workloads.

Deliverables:

- Deploy the backlog template.
- Update the template to reflect the first 10 workloads to be migrated.
- Update people and velocity to estimate release timing.
- Timeline risks:
 - Lack of familiarity with Azure DevOps can slow the deployment process.
 - Complexity and data available for each workload can also affect timelines.

Accountable team	Responsible and supporting teams
• Cloud adoption team	• Cloud strategy team

Step 6: Build a skills readiness plan

Existing employees can play a hands-on role in the migration effort, but additional skills might be required. In this step, find ways to develop those skills or use partners to add to those skills.

- [Build a skills-readiness plan](#). Quickly evaluate your existing skills to identify what other skills the team should develop.

Deliverables:

- Add a skills-readiness plan to the [strategy and plan template](#).

Accountable team	Responsible and supporting teams
• Cloud adoption team	• Cloud strategy team

Step 7: Deploy and align a landing zone

All migrated assets are deployed to a landing zone. The landing zone starts simple to support smaller workloads, then scales to address more complex workloads over time.

- [Choose a landing zone](#): Find the right approach to deploying a landing zone based on your adoption pattern. Then deploy that standardized code base.
- [Expand your landing zone](#): Whatever your starting point, identify gaps in the deployed landing zone and add required components for resource organization, security, governance, compliance, and operations.

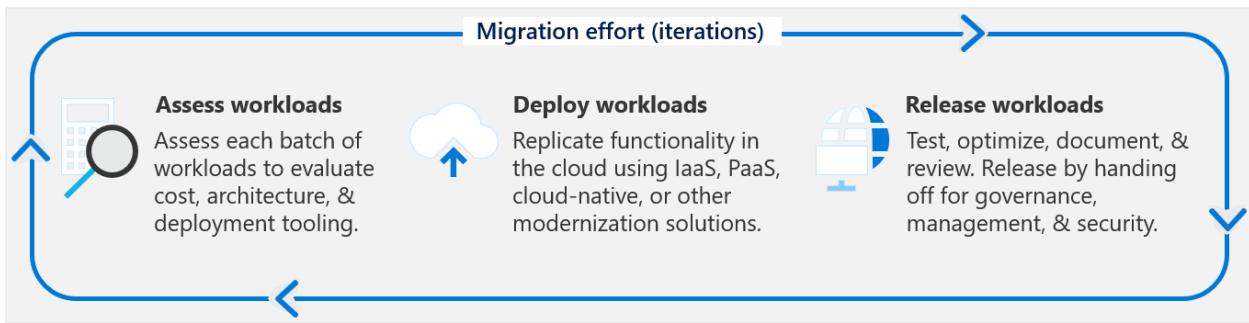
Deliverables:

- Deploy your first landing zone for deploying initial low-risk migrations.
- Develop a refactoring plan with the cloud center of excellence or the central IT team.
- Timeline risks:
 - Governance, operations, and security requirements for the first 10 workloads can slow this process.
 - Refactoring the first landing zone and subsequent landing zones takes longer, but it should happen in parallel with migration efforts.

Accountable team	Responsible and supporting teams
• Cloud platform team	• Cloud adoption team • Cloud center of excellence or central IT team

Step 8: Migrate your first 10 workloads

The technical effort required to migrate your first 10 workloads is relatively straightforward. It's also an iterative process that you repeat as you migrate more assets. In this process, you assess your workloads, deploy your workloads, and then release them to your production environment.



Cloud migration tools enable migrating all virtual machines in a datacenter in one pass or iteration. It's more common to migrate a smaller number of workloads during each iteration. Breaking up the migration into smaller increments requires more planning, but it reduces technical risks and the impact of organizational change management.

With each iteration, the cloud adoption team gets better at migrating workloads. These steps help the technical team mature their capabilities:

1. Migrate your first workload in a pure infrastructure as a service (IaaS) approach by using the tools outlined in the [Azure migration guide](#).
2. Expand tooling options to use migration and modernization by using the [migration examples](#).
3. Develop your technical strategy by using broader approaches outlined in [Azure cloud migration best practices](#).
4. Improve consistency, reliability, and performance through an efficient migration-factory approach as outlined in [Migration process improvements](#).

Deliverables:

Continuous improvement of the adoption team's ability to migrate workloads.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud adoption team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud center of excellence or central IT team

Step 9: Hand off production workloads to cloud governance

Governance is a key factor to the long-term success of any migration effort. Speed to migration and business impact is important. But speed without governance can be dangerous. Your organization needs to make decisions about governance that align to your adoption patterns and your governance and compliance needs.

- **Governance approach:** This methodology outlines a process for thinking about your corporate policy and processes. After determining your approach, you can build the disciplines required to enable governance across your enterprise cloud adoption efforts.
- **Initial governance foundation:** Understand the disciplines needed to create a governance minimum viable product (MVP) that serves as the foundation for all adoption.
- **Governance benchmark:** Identify gaps in your organization's current state of governance. Get a personalized benchmark report and curated guidance on how to get started.

Deliverables:

- Deploy an initial governance foundation.
- Complete a governance benchmark to plan for future improvements.
- Timeline risk:
 - Improvement of policies and governance implementation can add one to four weeks per discipline.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud governance team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud center of excellence or central IT team

Step 10: Hand off production workloads to cloud operations

Operations management is another requirement to reach migration success. Migrating individual workloads to the cloud without an understanding of ongoing enterprise operations is a risky decision. In parallel with migration, you should start planning for longer-term operations.

- **Define business commitments**
- **Establish a management baseline**
- **Expand the management baseline**
- **Advanced operations and design principles**

Deliverables:

- Deploy a management baseline.

- Complete the operations management workbook.
- Identify any workloads that require a Microsoft Azure Well-Architected Review assessment.
- Timeline risks:
 - Review the workbook: estimate one hour per application owner.
 - Complete the Microsoft Azure Well-Architected Review assessment: estimate one hour per application.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud operations team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud center of excellence or central IT team

Value statement

These steps help teams accelerate their migration efforts through better change management and stakeholder alignment. These steps also remove common blockers and realize business value more quickly.

Next steps

The Cloud Adoption Framework is a lifecycle solution that helps you begin a migration journey. It also helps mature the teams that support migration efforts. The following teams can use these next steps to continue to mature their capabilities. These parallel processes aren't linear and shouldn't be considered blockers. Instead, each is a parallel value stream to help improve your organization's overall cloud readiness.

Team	Next iteration
Cloud adoption team	Use the migration model to learn about moving toward a migration factory that provides efficient ongoing migration capabilities.
Cloud strategy team	Iteratively improve the Strategy methodology and the Plan methodology along with the adoption plan. Review these overviews and continue iterating on your business and technical strategies.
Cloud platform team	Revisit the Ready methodology to continue to advance the overall cloud platform that supports migration or other adoption efforts.
Cloud governance team	Use the Govern methodology to continue to improve governance processes, policies, and disciplines.

Team	Next iteration
Cloud operations team	Build on the Manage methodology to provide richer operations in Azure.

If your migration scenario is atypical, you can get a personalized assessment of your organization's migration readiness by using the [strategic migration and readiness tool \(SMART\) assessment](#). The answers you provide help identify which guidance aligns best with your current needs.

Get started: Accelerate new product and service innovation in the cloud

Article • 07/03/2023

Creating new products and services in the cloud requires a different approach than migration requires. The Innovate methodology of the Cloud Adoption Framework establishes an approach that guides the development of new products and services.

Innovation is less predictable than a standard migration, but it still fits within the context of the broader cloud adoption plan. This guide can help your enterprise provide the support needed to innovate and provide a structure for creating a balanced portfolio throughout cloud adoption.

Step 1: Document the business strategy

To avoid common blockers, create a clear and concise business strategy for innovation. Stakeholder alignment on motivations and expected business outcomes shapes decisions that the cloud adoption team makes.

Deliverables:

- Use the [strategy and plan template](#) to record motivations and desired business outcomes.

Guidance to support deliverable completion:

- **Motivations:** The first step to strategic alignment is to gain agreement on the motivations that drive the innovation effort. Start by understanding and categorizing motivations and common themes from stakeholders across business and IT.
- **Business outcomes:** After motivations are aligned, it's possible to capture the desired business outcomes. This information provides clear metrics that you can use to measure the overall transformation.
- **Balancing the portfolio:** Innovation isn't the right adoption path for every workload. This approach to adoption is more relevant to new custom-built applications or workloads that *require* rearchitecture or full rebuilds. When motivations heavily favor innovation for all workloads, it's important to evaluate the portfolio to ensure that those investments can produce the desired return on investment. Modernization of specific resources and small-scale rebuilding efforts

can be innovative but might be better served by following [Get started: Accelerate migration](#).

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">Cloud strategy team	<ul style="list-style-type: none">Cloud adoption teamCloud center of excellence or central IT team

Step 2: Evaluate the business justification

In this first pass at building the business case, evaluate the initial high-level return from a potential cloud adoption effort. The goal of this step is to align all stakeholders around one simple question: based on the available data, is the overall adoption of the cloud a wise business decision? Building on that question, the team can better align on how this innovation project helps meet the users' projected needs within the goal of adopting the cloud.

Deliverables:

- Use the [strategy and plan template](#) ↗ to record the business justification.

Guidance to support deliverable completion:

- Business justification:** Before you evaluate each opportunity to innovate in the cloud, complete a high-level business justification to establish stakeholder alignment for the overall adoption plan.
- Business value consensus:** Quantifying the value of an innovation can be difficult early in the process. The exercise in this article can aid in evaluating alignment on the business value of a specific innovation effort.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">Cloud strategy team	<ul style="list-style-type: none">Cloud adoption team

Step 3: Gather data and analyze assets and workloads

In most enterprises, innovation can be accelerated through the use of existing assets like applications, virtual machines (VMs), and data. When you plan for innovation, it's

important to understand how and when those assets are migrated to the cloud.

Deliverables:

- Get raw data on existing inventory like applications, VMs, and data.
- If the proposed innovation has dependencies on existing inventory, complete the following deliverables:
 - Quantitative analysis on any supporting inventory required to support the planned innovation.
 - Qualitative analysis of any supporting workloads required to deliver the innovation.
- Calculate the cost of new inventory required to support the innovation effort.
- Update the business justification in the [strategy and plan template](#) ↗ with refined calculations.

Guidance to support deliverable completion:

Discovery and assessment provide a deeper level of technical alignment. You can then create an action plan for migrating any dependent workloads that the planned innovation requires. This scenario is common when companies have existing data sources, centralized applications, or service layers that are necessary for delivering innovation within the context of the rest of the enterprise.

When there are dependent systems, the following articles can guide the discovery and assessment:

- [Inventory existing systems](#): Understanding the current state from a programmatic, data-driven approach is the first step. Discover and gather data to enable all assessment activities.
- [Incremental rationalization](#): Streamline assessment efforts to focus on a qualitative analysis of all assets, possibly even to support the business case. Then add a deep qualitative analysis for the first 10 workloads.

Accountable team	Responsible and supporting teams
• Cloud adoption team	• Cloud strategy team

Step 4: Plan for migration of dependent assets

When new innovation depends on existing workloads or assets, a cloud adoption plan provides an accelerated approach to developing a project backlog. The backlog can

then be modified to reflect discovery results, rationalization, needed skills, and partner contracting.

Deliverables:

- Deploy the backlog template.
- Update the template to reflect the first 10 workloads to be migrated.
- Update people and velocity (people's time) to estimate release timing.
- Timeline risks:
 - Lack of familiarity with Azure DevOps can slow the deployment process.
 - Complexity and data available for each workload can also affect timelines.

Guidance to support deliverable completion:

- [Cloud adoption plan](#): Define your plan using the basic template.
- [Workload alignment](#): Define workloads in the backlog.
- [Effort alignment](#): Align assets and workloads in the backlog to clearly define the effort for prioritized workloads.
- [People and time alignment](#): Establish iteration, velocity, and releases for the workloads.

Accountable team	Responsible and supporting teams
• Cloud adoption team	• Cloud strategy team

Step 5: Align governance requirements to your adoption plan

Discussing planned innovations with the governance team helps you avoid many blockers before they arise. Sometimes, innovative new solutions might require practices that are discouraged in sound governance practices. Some of those required features might even be blocked through automated tooling for governance enforcement.

Deliverables:

- Create transparency and understanding between innovation needs and governance constraints.
- When necessary, update policies and processes to reflect any changes or exceptions to existing governance constraints.

Guidance to support deliverable completion:

These links help the adoption team understand the approach of the cloud governance team:

- [Governance approach](#): This methodology outlines a process for thinking about corporate policy and processes. Then you can build the disciplines required to deliver on governance across your cloud enterprise efforts.
- [Definition of corporate policy](#): Identify and mitigate business risks.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud governance team• Cloud adoption team	<ul style="list-style-type: none">• Cloud strategy team• Cloud center of excellence or central IT team

Step 6: Define operational needs and business commitments

Define the plan for long-term operational responsibilities for the planned innovation. Will the established management baseline meet your operational needs? If not, evaluate options for funding operations that are specific to the technology that supports this innovation.

Deliverables:

- Complete the [Microsoft Azure Architecture Review](#) to assess various architecture and operation decisions.
- Adjust the [operations management workbook](#) ↗ to reflect any required advanced operations.

Guidance to support deliverable completion:

- [Expand the management baseline](#): This section of the Cloud Adoption Framework guides you through various transitions into operational management in the cloud.
- [Get specific with advanced operations](#): Discover ways to go beyond your management baseline.
- If advanced operations are required to support your operations needs, evaluate the [business commitments](#) to determine operational responsibilities for both teams.

Accountable team	Responsible and supporting teams
------------------	----------------------------------

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud operations team • Cloud adoption team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud center of excellence or central IT team

Step 7: Deploy an aligned landing zone

All assets hosted in the cloud live within a landing zone. That landing zone might have explicit governance, security, and operational requirements. Or, it might be a new subscription without support from other teams. In either scenario, it's important to start with a landing zone that aligns to governance and operational requirements from the beginning.

Starting with an approved landing zone helps your team to discover policy violations early during development versus when the solution is released to production. Early discovery helps your team to remove blockers and gives adoption and governance teams enough time to make changes.

Deliverables:

- Deploy a first landing zone for initial, low-risk experimentation during early innovation.
- Develop a plan to refactor with the cloud center of excellence or the central IT team to ensure governance, security, and operational alignment.
- Timeline risks:
 - Governance, operations, and security requirements for the first 10 workloads can slow this process. Refactoring the first landing zone and later landing zones takes longer, but it should happen in parallel with migration efforts.

Guidance to support deliverable completion:

- [Choose a landing zone](#): Use this section to find the right approach to deploying a landing zone based on your adoption pattern. Then deploy that standardized code base.
- [Expand your landing zone](#): Regardless of the starting point, identify gaps in the deployed landing zone to add required components for resource organization, security, governance, compliance, and operations.

Accountable team	Responsible and supporting teams
------------------	----------------------------------

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud platform team • Cloud adoption team 	<ul style="list-style-type: none"> • Cloud adoption team • Cloud center of excellence or central IT team

Step 8: Innovate in the cloud

The Innovate methodology provides guidance on the tools and product management approaches most commonly used to innovate in the cloud. These steps help you get started with this approach.

Deliverables:

- Technology-based solutions that enrich your customers' lives and drive value for the business.
- Processes and tools to iterate on those solutions faster and add more value by using the cloud:
 - Iterative development approaches.
 - Custom-built applications.
 - Technology-based experiences.
 - Integration of physical products and technology by using IoT.
 - Ambient intelligence: integration of nonintrusive technology into an environment.
 - Azure Cognitive Services: big data, AI, machine learning, and predictive solutions.

Guidance to support deliverable completion:

- **Create hypothesis with business value consensus:** Before you decide on technical solutions, identify how new innovations can drive business value and come up with a hypothesis about customer needs.
- **Build your first MVP:** Once a hypothesis has enough value potential to be built into your application, the build process starts. Development sprints should be as quick as possible, to allow quick verification or rejection of the hypothesis, or to fine tune the way in which the required functionality is to be integrated in the application.
- **Measure and learn:** You want to verify the accuracy of your hypothesis as soon as possible. A minimum viable product (MVP) is a preliminary version of the new functionality, that offers just enough functionality to gather feedback that confirms if you are moving in the right direction.
- **Expand digital innovation:** To refine your hypothesis using the innovation disciplines or the digital inventions which include democratize data, engage via

applications, empower adoption, interact with devices, and predict and influence.

These inventions are a core part of the Innovate methodology.

Step 9: Assess the innovation maturity of your organization

To support the development of your innovation strategy, the [AI readiness assessment tool](#) is a free resource that helps organizations assess their ability to create and own AI-based systems. There are four levels of maturity: foundational, approaching, aspirational, and mature. Each level includes a specific set of characteristics to help determine your organization's ability to adopt specific types of AI solutions, mitigate associated risks, and implement strategies.

The assessment takes 5 to 10 minutes and measures your organization's capability across four categories: strategy, culture, organizational characteristics, and capabilities. Measuring these categories allows the AI readiness assessment tool to compute your organization's score and provide an estimate of the AI innovation maturity on a curve.

Deliverables:

- Use the [Gartner AI Maturity Model](#) to assess your organization's AI maturity to create AI-based systems.

Guidance to support deliverable completion:

- When the assessment is complete, the tool's output will provide a score that estimates the status of AI innovation maturity.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud adoption team	<ul style="list-style-type: none">• Cloud center of excellence• Cloud center of excellence or central IT team

Value statement

The steps outlined in this guide can help you and your teams create innovative solutions in the cloud that create business value, are governed appropriately, and are well architected.

Next steps

The Cloud Adoption Framework is a lifecycle solution. It can help you begin an innovation journey. It can help your organization to start an innovation journey and to advance the maturity of the teams that support innovation efforts.

The following teams can use these next steps to continue to advance the maturity of their efforts. These parallel processes aren't linear and shouldn't be viewed as blockers. Instead, each is a parallel value stream to help mature your company's overall cloud readiness.

Team	Next iteration
Cloud adoption team	Process improvements provide insight about approaches to deliver on innovations that affect customers and drive recurring adoption.
Cloud strategy team	The Strategy methodology and the Plan methodology are iterative processes that evolve with the adoption plan. Return to these overview pages and continue to iterate on your business and technical strategies.
Cloud platform team	Revisit the Ready methodology to continue to advance the overall cloud platform that supports migration or other adoption efforts.
Cloud governance team	Use the Govern methodology to continue to improve governance processes, policies, and disciplines.
Cloud operations team	Build on the Manage methodology to provide richer operations in Azure.

Get started: Environment design and configuration

Article • 02/28/2023

Environment design and configuration are the most common blockers to adoption efforts that are focused on migration or innovation. Quickly implementing a design that supports your long-term adoption plan can be difficult. This article establishes an approach and series of steps that help to overcome common blockers and accelerate your adoption efforts.

The technical effort required to create an effective environmental design and configuration can be complex. You can manage the scope to improve the odds of success for the cloud platform team. The greatest challenge is alignment among multiple stakeholders. Some of these stakeholders have the authority to stop or slow the adoption efforts. These steps outline ways to quickly meet short-term objectives and establish long-term success.

Step 1: Document the business strategy

To avoid common migration blockers, make sure that you have a clear and concise business strategy. Stakeholder alignment on motivations, expected business outcomes, and the business justification is important throughout adoption and environment configuration.

A clear and concise business strategy helps the cloud platform team understand what's important and what should be prioritized when they're making environmental configuration decisions. In particular, it helps the teams make decisions when they're forced to choose between speed of innovation or adherence to controls.

Deliverables:

- Use the [strategy and plan template](#) ↗ to record motivations, desired business outcomes, and high-level business justification.

Guidance to support deliverable completion:

- **Understand business motivations:** The first step to strategic alignment is to agree on the motivations that drive the migration effort. Start by understanding and categorizing motivations and common themes from various stakeholders across business and IT.

- **Document business outcomes:** After motivations are aligned, you can capture the desired business outcomes. This information provides clear metrics you can use to measure the overall transformation.
- **Build a cloud migration business case:** Start developing a business case for migration, including clear guidance on the formulas and tools that help your business justification.

Accountable team	Responsible and supporting teams	Informed teams
<ul style="list-style-type: none"> • Cloud strategy team 	<ul style="list-style-type: none"> • Cloud adoption team • Cloud center of excellence or central IT team 	<ul style="list-style-type: none"> • Cloud platform team

Step 2: Assess the digital estate

Discovery and assessment provide a deeper level of technical alignment, which helps you create an action plan you can use to deliver on the strategy. During this step, you validate the business case by using data about the current state of the environment. Then you perform quantitative analysis of that data and a deep qualitative assessment of the highest priority workloads.

The output of the digital estate assessment provides the cloud platform team with a clear view of the end-state environment and the requirements that are needed to support the adoption plan.

Deliverables:

- Raw data on the existing inventory.
- Quantitative analysis of the existing inventory to refine the business justification.
- Qualitative analysis of the first 10 workloads.
- Updated business justification in the [strategy and plan template ↗](#).

Guidance to support deliverable completion:

- **Inventory existing systems:** Understanding the current state from a programmatic, data-driven approach is the first step. Find and gather data to enable all assessment activities.
- **Incremental rationalization:** Streamline assessment efforts to focus on a qualitative analysis of all assets, possibly even to support the business case. Then add a deep qualitative analysis for the first 10 workloads to be migrated.

Accountable team	Responsible and supporting teams	Informed teams
• Cloud adoption team	• Cloud strategy team	• Cloud platform team

Step 3: Create a cloud adoption plan

Your cloud adoption plan provides an accelerated approach to developing a project backlog. The backlog can then be modified to reflect assessment results, rationalization, needed skills, and partner contracting.

A review of the short-term cloud adoption plan and backlog helps the cloud platform team understand the needs of the environment for the next few months. This background helps them to tighten their *definition of done* for the first few landing zones.

Deliverables:

- Deploy the backlog template.
- Update the template to reflect the first 10 workloads to be migrated.
- Update people and velocity (people's time) to estimate release timing.
- Timeline risks:
 - Lack of familiarity with Azure DevOps can slow the deployment process.
 - Complexity and data available for each workload can also affect timelines.

Guidance to support deliverable completion:

- [Cloud adoption plan](#): Define your plan using the basic template.
- [Workload alignment](#): Define workloads in the backlog.
- [Effort alignment](#): Align assets and workloads in the backlog to clearly define efforts for prioritized workloads.
- [People and time alignment](#): Establish iteration, velocity, and releases for the migrated workloads.

Accountable team	Responsible and supporting teams	Informed teams
• Cloud adoption team	• Cloud strategy team • Cloud platform team	• Cloud platform team

Step 4: Deploy the first landing zone

Initially, the cloud adoption team needs a landing zone that can support the requirements of the first wave of workloads. Over time, the landing zone scales to address more complex workloads. For now, start with a landing zone that enables early learning for the cloud platform team and the cloud adoption team.

Deliverables:

- Deploy a first landing zone for initial low-risk migrations.
- Develop a plan to refactor with the cloud center of excellence or the central IT team.
- Timeline risks:
 - Governance, operations, and security requirements for the first 10 workloads can slow this process. Actual refactoring of the first landing zone and subsequent landing zones takes longer, but it should happen in parallel with migration efforts.

Guidance to support deliverable completion:

- [Choose a landing zone](#): Use this section to find the right approach to deploying a landing zone based on your short-term adoption plan. Then deploy that standardized code base.
- [Expand your landing zone](#): Don't attempt to meet long-term governance, security, or operation constraints yet, unless they're required to support the short-term adoption plan.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud platform team	<ul style="list-style-type: none">• Cloud adoption team• Cloud center of excellence or central IT team

Step 5: Deploy an initial governance foundation

Governance is a key factor to the long-term success of any migration effort. Speed to migration and business impact is important. But speed without governance can be dangerous. Your organization needs to make decisions about governance that align to your adoption patterns and your governance and compliance needs.

As those decisions are made, they feed back into the parallel efforts of the cloud platform team.

Deliverables:

- Deploy an initial governance foundation.
- Complete a governance benchmark to plan for future improvements.
- Timeline risks:
 - Improvement of policies and governance implementation can add one to four weeks per discipline.

Guidance to support deliverable completion:

- [Governance approach](#): This methodology outlines a process for thinking about corporate policy and processes. Then build the disciplines required to deliver on governance across your cloud enterprise adoption efforts.
- [Governance benchmark tool](#): Find gaps in your current state so that you can plan for the future.
- [Initial governance foundation](#): Understand the governance disciplines that are required to create a governance minimum viable product (MVP) to serve as the foundation for all adoption.

Accountable team	Responsible and supporting teams	Consulted teams
<ul style="list-style-type: none"> • Cloud governance team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud center of excellence or central IT team 	<ul style="list-style-type: none"> • Cloud platform team

Step 6: Implement an operations baseline

Migrating to the cloud without understanding ongoing operations is risky. In parallel with migration, start planning for longer-term operations management. Feed those plans back into the parallel efforts of the cloud platform team.

Deliverables:

- Deploy a management baseline.
- Complete the operations management workbook.
- Identify any workloads that require a Microsoft Azure Well-Architected Review assessment.
- Timeline risks:
 - Review the workbook: estimate one hour per application owner.
 - Complete the Microsoft Azure Well-Architected Review assessment: estimate one hour per application.

Guidance to support deliverable completion:

- Establish a management baseline
- Define business commitments
- Expand the management baseline
- Get specific with advanced operations

Accountable team	Responsible and supporting teams	Consulted teams
<ul style="list-style-type: none"> • Cloud operations team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud center of excellence or central IT team 	<ul style="list-style-type: none"> • Cloud platform team

Step 7: Expand the landing zone

As the cloud adoption team begins their first few migrations, the cloud platform team can begin building toward the end-state environment configuration with the support of the cloud governance and cloud operations teams. Depending on the pace of the cloud adoption plan, this process might need to happen in iterative releases. Functionality might be added ahead of the requirements of the adoption plan.

Deliverables:

- Adopt a test-driven development approach to refactoring landing zones.
- Improve landing zone governance.
- Expand landing zone operations.
- Implement landing zone security.

Guidance to support deliverable completion:

- Refactor landing zones
- Test-driven development of landing zones
- Expand landing zone governance
- Expand landing zone operations
- Expand landing zone security

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud platform team 	<ul style="list-style-type: none"> • Cloud adoption team • Cloud center of excellence or central IT team

Value statement

The steps outlined in this guide can help you and your teams accelerate their path to an enterprise-ready cloud environment that's properly configured.

Next steps

Consider these next steps in a future iteration to build on your initial efforts:

- [Environmental technical readiness learning paths](#)
- [Migration environment planning checklist](#)

Enable customer success with a sound operating model and organizational alignment

Article • 02/28/2023

Customer success in cloud adoption efforts often has little to do with technical skills or adoption-related projects. Your operating model creates opportunities to enable adoption or roadblocks that might slow down cloud adoption.

Alignment

As you drive innovation, alignment between business and technical teams is paramount to the success of your solution.

- For business stakeholders, we've created the [Microsoft AI Business School](#) to support business strategy development and provide example best practices.
- For technical stakeholders, the [Microsoft AI learning paths](#) are available to help you build new AI skills.

Blockers

When adoption of the cloud is slowed or stalled, it might be wise to evaluate your operating model to enable continued success. When success is inconsistent from workload to workload or project to project, the operating model might be misaligned. If more than one project is stalled by blocking policies, outdated processes, or misalignment of people, the operating model is likely blocking success.

Opportunities

Beyond the common blockers, a few key opportunities can be scaled across the portfolio through incremental improvements to your operating model. In particular, customers commonly want to scale operational excellence, cost optimization, security, reliability, performance, or people management. Scaling these conversations at the portfolio level can help bring best practices for specific workload-focused teams to all other projects and workloads.

Getting started guides to enable teams through an operating model

The following guides will help you get started with operating model alignment and improve over time.

Guide	Description
How do we deliver operational excellence during cloud transformation?	The steps in this guide will help the strategy team lead organizational change management to consistently ensure operational excellence.
How do we manage enterprise costs?	Start optimizing enterprise costs and manage cost across the environment.
How do we consistently secure the enterprise cloud environment?	This getting started guide can help ensure that the proper security requirements have been applied across the enterprise to minimize risk of breach and accelerate recovery when a breach occurs.
How do we apply the right controls to improve reliability?	This getting started guide helps minimize disruptions related to inconsistencies in configuration, resource organization, security baselines, or resource protection policies.
How do we ensure performance across the enterprise?	This getting started guide can help you establish processes for maintaining performance across the enterprise.
How do we align our organization?	This getting started guide can help you establish an appropriately staffed organizational structure.

Shared architecture principles

The core principles of a well-managed operating model are based on a set of common architecture principles. The getting started guidance in this article series will help supporting teams as they scale these principles across the cloud platform and throughout the portfolio of workloads.



Figure 1: The principles of shared architecture.

These principles are shared across Azure Advisor, the Microsoft Azure Well-Architected Framework, and solutions in the Azure Architecture Center:

- [Azure Advisor](#) evaluates the principles for individual assets across solutions, workloads, and the full portfolio.
- [Azure Architecture Center](#) applies these principles to develop and manage specific technical solutions.
- [Microsoft Azure Well-Architected Framework](#) helps balance these principles across a workload, to guide architecture decisions.
- [Cloud Adoption Framework](#) ensures that the principles scale across the portfolio to enable adoption teams through a well-managed environment.

Get started: Deliver operational excellence during digital transformation

Article • 02/28/2023

How do you ensure operational excellence during digital transformation? Operational excellence is a business function that directly affects IT decisions. To achieve operational excellence, you must focus on customer and stakeholder value by keeping an eye on revenue, risk, and cost impacts.

This organizational change management approach requires:

- A defined strategy.
- Clear business outcomes.
- Change management planning.

From a cloud perspective, you can manage the impact of risk and cost by making post-adoption changes and continuously refining operational processes. Areas to monitor include systems automation, IT operations management practices, and Resource Consistency discipline throughout the cloud adoption lifecycle.

The steps in this article can help the strategy team lead the organizational change management that's required to consistently ensure operational excellence.

Operational excellence across the enterprise and portfolio starts with peer processes of strategy and planning to align and report on organizational change management expectations. The following steps help technical teams deliver the disciplines required to achieve operational excellence.

Step 1: Define a strategy to guide digital transformation and operational excellence expectations

A clear business strategy is the foundation for any digital transformation and operational excellence effort. IT can reduce costs and streamline IT processes. Without a clear strategy, it's difficult to understand how those changes might affect the business outcomes identified in the broader transformation effort.

Deliverables:

- Record motivations, outcomes, and business justification in the [strategy and plan template ↗](#).
- Ensure learning metrics are well understood and included in the business outcomes section. Those metrics guide operational excellence activities and reporting within IT.

Guidance to support deliverable completion:

- **Understand motivations:** Critical business events and some migration motivations tend to be cost sensitive. These areas can increase the importance of cost control for all later efforts. Other forward-looking motivations related to innovation or growth through migration might be focused more on top-line revenue. Understanding motivations helps you prioritize your cost management.
- **Business outcomes:** Some fiscal outcomes tend to be extremely cost sensitive. When the desired outcomes map to fiscal metrics, you should invest early in your Cost Management discipline.
- **Business justification:** The business justification serves as a high-level view of the overall financial plan for cloud adoption. It can be a good source for initial budgeting efforts.
- **Learning metrics:** To maintain alignment between the overarching business strategy and the more tactical change-management plans, establish learning metrics. These metrics should be designed to show iterative and incremental progress toward the plan.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud strategy team 	<ul style="list-style-type: none"> • Cloud adoption team • Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 2: Develop an organizational change management plan to span cloud adoption

Organizational change management is an iterative approach to subtly realign people, processes, and technology to support a holistic business strategy. In the case of operational excellence for digital transformation, this approach often centers on an IT-centric cloud adoption plan.

Deliverables:

- Update the [strategy and plan template](#) to reflect change that's needed to achieve the desired strategy. The changes recorded can include:
 - An assessment of the existing digital estate.
 - A cloud adoption plan that reflects the required changes and the work involved.
 - The organizational changes that are required to deliver on the plan.
 - A plan for addressing the skills that are needed to enable the existing team to successfully complete the required work.

Guidance to support deliverable completion:

- **Gather inventory:** Establish a source of data for analysis of the digital estate prior to adoption.
- **Best practice: Azure Migrate:** Use Azure Migrate to gather inventory.
- **Incremental rationalization:** During incremental rationalization, a quantitative analysis identifies cloud candidates for budgeting purposes.
- **Align cost models and forecast models:** Use Azure Cost Management + Billing to align cost and forecast models by [creating budgets](#).
- **Build your cloud adoption plan:** Build a plan with actionable workload, assets, and timeline details.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud strategy team 	<ul style="list-style-type: none"> • Cloud adoption team • Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 3: Manage change across cloud adoption efforts

Realization of business outcomes is the result of continuous delivery of adoption waves. Those waves could include migration and innovation cycles. In either case, delivery on operational excellence starts with regular cycles of change management.

Each wave (or release, in agile terms) delivers a set of workloads to the cloud. As each wave of adoption is completed, the cloud strategy team reports on progress toward learning metrics, business outcomes, and the overall strategy. Likewise, as each wave of adoption is completed, the adoption teams need backlog updates that reflect the prioritized workloads in the plan. These updates are based on any changes to business plans and customer needs.

Deliverables:

- Continuous testing and improvements to the strategy and change management plan based on changing market conditions and completion of the most recent wave of technical change.

Guidance to support deliverable completion:

- **Release planning:** Approaches to change management through release planning.
- **Incremental rationalization:** Iterative approach to change management. The focus is on managing the release backlog to support manageable waves of change.
- **Power of 10 approach:** Limits the change management plan. The focus is on detailed analysis and planning of a continuous base of 10 workloads to balance incremental change and iterative adoption efforts.
- **Align iteration paths:** Update and add details at each release to ensure current iteration paths.
- **Assess workloads:** The efforts of the cloud adoption team to evaluate and act on the most recent set of migration priorities.
- **Business value consensus:** The cloud adoption team's efforts to ensure business value alignment at each release of new innovation.

Accountable team	Responsible and supporting teams
• Cloud strategy team	• Cloud adoption team

Value statement

The previous steps outline a business-led approach to establish operational excellence requirements throughout digital transformation. This approach provides a consistent foundation that carries through other operating model functions.



Next steps to delivering operational excellence across the portfolio

Operational excellence requires a disciplined approach to reliability, performance, security, and cost optimization. Use the remaining guidance in this series to implement these principles through consistent approaches to automation.

- **Cost optimization:** Continuously optimize operating costs by using the getting started guide on [managing enterprise costs](#)
- **Security:** Reduce risk by integrating enterprise security across the portfolio by using the getting started guide on [implementing security across the portfolio](#).
- **Performance management:** Ensure IT asset performance supports business processes by using the getting started guide on [performance management across the enterprise](#).
- **Reliability:** Improve reliability and reduce business disruptions by using the getting started guide on [implementing controls to create reliability](#).

Get started: Manage cloud costs

Article • 02/28/2023

The Cost Management discipline of cloud governance focuses on establishing budgets, monitoring cost allocation patterns, and implementing controls to improve cloud spending behaviors across the IT portfolio.

However, enterprise cost optimization involves many other roles and functions to minimize cost and balance the demands of scale, performance, security, and reliability. This article maps those supporting functions to help create alignment between the involved teams.

Governance is the cornerstone of cost optimization within any large enterprise. The following section outlines cost optimization guidance within the context of governance. The subsequent steps help each team take actions that target its role in cost optimization. Together, these steps will help your organization get started on a journey toward cost optimization.

Step 1: Optimize enterprise costs

The cloud governance team is well prepared to evaluate and act on overspending or unplanned spending through a combination of monitoring performance, reducing resource sizing, and safely terminating unused resources. Enterprise cost optimization starts with a shared team understanding of the tools, processes, and dependencies required to wisely act on cost concerns at an environment level.

Deliverables:

- Implement wise changes to your Cost Management policies across the enterprise.
- Document your Cost Management policies, processes, and design guidance in the [Cost Management discipline template](#).

These deliverables are the result of a few recurring tasks:

- Ensure strategic alignment with the cloud strategy team (which includes workload stakeholders across the portfolio).
- Optimize cost across the environment:
 - Manually or automatically shut down unused VMs.
 - Delete or deallocate stopped VMs.
 - Ensure proper resource sizing.
 - Align spending to budget expectations.

- Validate any architectural change by using the Microsoft Azure Well-Architected Review to facilitate a conversation with technical owners of the workloads.

Guidance to support deliverable completion:

- Ensure that all workloads and resources follow [proper naming and tagging conventions](#). Enforce tagging conventions by using [Azure Policy](#) with a specific emphasis on tags for `cost center` and `technical owner`.
- On a regular basis, review and apply [Cost Management discipline best practices](#) to guide analysis and improvements across the enterprise. Important governance practices include:
 - Acting on [general cost best practices](#) to reduce sizing and costs and to stop unused machines.
 - Applying [hybrid use benefits](#) to reduce licensing costs.
 - Aligning [reserved instances](#) to reduce resource costs.
 - [Monitoring resource utilization](#) to minimize impacts on resource performance.
 - [Reducing nonproduction costs](#) through policies to govern nonproduction environments.
 - Acting on [cost optimization recommendations](#).
- Trade-offs at the workload level might be needed to implement effective cost optimization changes. The [Microsoft Azure Well-Architected Framework](#) and [Microsoft Azure Well-Architected Review](#) can help guide those conversations with the technical owner of a specific workload.
- If you're new to cloud governance, establish [governance policies, processes, and disciplines](#) using the [Govern](#) methodology.
- If you're new to the Cost Management discipline, consider following the [Cost Management discipline improvements article](#), with a focus on the [Implementation](#) section.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud governance team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud adoption team • Cloud center of excellence or central IT team

The governance team can detect and drive significant cost optimization across most enterprises. Basic, data-driven resource sizing can have an immediate and measurable impact on costs.

As discussed in [Build a cost-conscious organization](#), an enterprise-wide focus on cost management and cost optimization can deliver much more value. The following steps demonstrate ways the various teams can help build a cost-conscious organization.

Step 2: Define a strategy

Strategic decisions directly affect cost controls, rippling through the adoption lifecycle and into long-term operations. Strategic clarity will improve cost optimization efforts, driven by the governance team.

Deliverables:

- Record motivations, outcomes, and business justification in the [strategy and plan template](#).
- Create your first budget by using Azure Cost Management + Billing.

Guidance to support deliverable completion:

- **Understand motivations.** Critical business events and some migration motivations tend to be cost sensitive, increasing the importance of cost control for all later efforts. Other forward-looking motivations related to innovation or growth through migration might focus more on top-line revenue. Understanding motivations will help you decide how high to prioritize your cost management.
- **Business outcomes.** Some fiscal outcomes tend to be extremely cost-sensitive. When the desired outcomes map to fiscal metrics, you should invest in the Cost Management governance discipline very early.
- **Business justification.** The business justification serves as a high-level view of the financial plan for cloud adoption. This is a good source for initial budgeting efforts.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud strategy team	<ul style="list-style-type: none">• Cloud governance team• Cloud adoption team• Cloud center of excellence or central IT team

Step 3: Develop a cloud adoption plan

The adoption plan provides clarity on the timeline of activities during adoption. Aligning the plan and the digital estate analysis allows you to forecast your monthly growth in

spending. It also helps your cloud governance team align processes and identify spending patterns.

Deliverables:

- Complete steps 1 through 6 of building a [cloud adoption plan](#).
- Work with your cloud governance team to refine budgets and create realistic spending forecasts.

Guidance to support deliverable completion:

- [Gather inventory](#). Establish a source of data for analysis of the digital estate before adoption.
- [Best practice: Azure Migrate](#). Use Azure Migrate to gather inventory.
- [Incremental rationalization](#). During incremental rationalization and quantitative analysis, identify cloud candidates for budgeting purposes.
- [Align cost models and forecast models](#). Use Azure Cost Management + Billing to align cost and forecast models by [creating budgets](#).
- [Build your cloud adoption plan](#). Build a plan with actionable workload, assets, and timeline details. This plan provides the basis for spending over time (or cost forecasting). *Spending over time* is the initial baseline for all actionable optimization analysis within the Cost Management governance discipline.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud adoption team	<ul style="list-style-type: none">• Cloud strategy team• Cloud governance team• Cloud center of excellence or central IT team

Step 4: Implement best practices for landing zones

The Ready methodology of the Microsoft Cloud Adoption Framework for Azure focuses heavily on the development of landing zones to host workloads in the cloud. During implementation of landing zones, an organization should consider various decisions for cost optimization.

Deliverables:

- Deploy one or more landing zones that can host workloads in the short-term adoption plan.

- Ensure that all landing zones meet cost optimization decisions and cost management requirements.

Guidance to support deliverable completion:

- [Track costs](#). Establish a well-managed environment hierarchy, provide the right level of cost access, and use additional cost management resources in each landing zone.
- [Optimize your cloud investment](#). Understand best practices for optimizing investments.
- [Create and manage budgets](#). Understand best practices for creating and managing budgets.
- [Optimize costs from recommendations](#). Understand best practices for using recommendations that will optimize costs.
- [Monitor usage and spending](#). Understand best practices for monitoring usage and spending within a landing zone.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud adoption team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 5: Complete waves of migration effort

Migration is a repeatable process executed by the cloud adoption team. Throughout this process, there are many opportunities to optimize costs across your portfolio. Many of these in-process decisions are applied to a small group of workloads during each migration wave or iteration.

Deliverables:

- Benchmark, test, resize, and deploy a collection of fully optimized workloads.

Guidance to support deliverable completion:

- [Migration-focused cost-control mechanisms](#) provides insights about the cloud-native cost optimization controls that help during migration.
- [Best practices for optimizing cost of migrated workloads](#) contains a checklist of 14 best practices to follow before and after migration to maximize cost optimization of each workload release.

Long-term operational costs are a common theme in each area of migration process improvements. This list of process improvements is organized by the phase of the migration process:

- [Prerequisites](#) provides information on managing change and the backlog, which influences both budgeted and actual cloud costs.
- [Assess](#) provides six specific processes, from validating assumptions to understanding partner options. Each process influences cloud optimization opportunities.
- [Migrate](#) contains one process suggestion about remediating assets. This suggestion provides an opportunity to optimize the as-configured state, in favor of an optimized solution.
- [Promote](#) focuses heavily on testing, resizing, validating, and releasing migrated assets, along with decommissioning retired assets. This is the first clear point at which forecasts and budgets can be tested against actual performance and configuration.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud adoption team	<ul style="list-style-type: none">• Cloud strategy team• Cloud governance team• Cloud center of excellence or central IT team

Step 6: Drive customer-focused innovation

Innovation and development of new products require a much deeper degree of architectural review. The Cloud Adoption Framework provides details on the innovation process and product management thinking. Cost optimization decisions about innovations are largely out of scope in this guidance.

Deliverables:

- Make key architectural decisions about innovations to balance cost and other critical design considerations.

Guidance to support deliverable completion:

- Use the [Microsoft Azure Well-Architected Review](#) to understand the balance in architecture decisions.
- Review the [Microsoft Azure Well-Architected Framework](#) for deeper guidance on cost optimization during innovation.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud adoption team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 7: Implement sound operations

Establishing a solid management baseline will help you collect data and create operational alerts. This effort can aid in detecting opportunities to optimize costs. It will create a balance between resiliency and cost optimization.

Deliverables:

- Monitor your enterprise environment for ongoing recommendations to optimize costs, aligned to the criticality and resiliency classifications of each workload.

Guidance to support deliverable completion:

- [Create business alignment](#) to gain clarity regarding criticality and appetite for resiliency investments.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud operations team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Value statement

Following these steps helps you [build a cost-conscious organization](#). Simplify cost optimization by using shared ownership and driving collaboration with the right teams at the right times.

Get started: Improve reliability with the right controls

Article • 02/28/2023

How do you apply the right controls to improve reliability? This article helps you minimize disruptions related to:

- Inconsistencies in configuration.
- Resource organization.
- Security baselines.
- Resource protection.

The steps in this article help the operations team balance reliability and cost across the IT portfolio. This article also helps the governance team to ensure that balance is applied consistently. Reliability also depends on other roles and functions. This article maps supporting functions to help you create alignment among the involved teams.

Operations management and governance are equal partners in enterprise reliability. The decisions you make about operational practices set the baseline for reliability. The approaches used to govern the overall environment ensure consistency across all resources.

The first two steps in this article help both teams get started. They're listed sequentially, but you can perform them in parallel. The subsequent steps help you get the entire enterprise started on a shared journey toward more reliable solutions throughout the enterprise.

Step 1: Establish operations management requirements

Not all workloads are created equal. In any environment, there are workloads that have a direct and constant impact on the business. There are also supporting business processes and workloads that have a smaller impact on the overall business. In this step, the cloud operations team identifies and implements initial requirements to support the overall IT portfolio.

Deliverables:

- Implement a management baseline to define the standard operations that are required for all production workloads.

- Negotiate business commitments with the cloud strategy team to develop a plan for advanced operations and resiliency requirements.
- Expand your management baseline, if additional operations are required for the majority of workloads.
- Apply advanced operations requirements to landing zones and resources that support the workloads that are most critical.
- Document operations decisions across the IT portfolio in the [operations management workbook](#).

Guidance to support deliverable completion:

- **Management baseline:**
 - **Inventory and visibility:** [Cloud-native tools](#) can help you [collect data](#) and [configure alerts](#). The tools also can help you implement the [monitoring platform](#) that best fits your operating model.
 - **Operational compliance:** The highest percentages of outages tend to come from changes to resource configuration or poor maintenance practices. Follow the [Azure server management guide](#) to implement cloud-native tools to manage patching and changes to resource configuration.
 - **Protection and recovery:** Outages are inevitable on any platform. When a disruption occurs, be prepared with [backup and recovery solutions](#) to minimize the duration.
- **Advanced operations:** Use the management baseline as the foundation for your [business alignment](#) conversations. It helps you to clearly discuss [criticality](#), [business impact](#), and [operations commitments](#). Business alignment helps quantify and validate requests for an [enhanced baseline](#), management of specific [technology platforms](#), or [workload-specific operations](#).
- **Guide an architecture review:** Architecture changes at the workload level might be required to meet operations requirements. The [Microsoft Azure Well-Architected Framework](#) and [Microsoft Azure Well-Architected Review](#) can help guide those conversations with the technical owner of a specific workload.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud operations team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud adoption team • Cloud governance team • Cloud center of excellence or central IT team

Step 2: Consistently apply the management baseline

Enterprise reliability requires consistent application of the management baseline. That consistency comes from appropriate corporate policy, IT processes, and automated tools. These resources govern the implementation of the management baseline for all affected resources.

Deliverable:

Ensure proper application of the management baseline for all affected systems.

Guidance to support deliverable completion:

- Ensure all workloads and resources follow [proper naming and tagging conventions](#). Enforce tagging conventions by using [Azure Policy](#), with a specific emphasis on tags for criticality.
- If you're new to cloud governance, establish [governance policies, processes, and disciplines](#) by using the Govern methodology.
- If you're new to the Cost Management discipline, follow the guidance in the [Cost Management discipline improvements](#) article. Focus on the [Implementation](#) section.

Note

Steps to start reliability partnerships with other teams: Various decisions throughout the cloud adoption lifecycle can have a direct impact on reliability. The following steps outline the partnerships and supporting efforts required to deliver consistent reliability across the IT portfolio.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud governance team	<ul style="list-style-type: none">• Cloud strategy team• Cloud operations team• Cloud center of excellence or central IT team

Step 3: Define your strategy

Strategic decisions directly affect reliability. They ripple through the adoption lifecycle and into long-term operations. Strategic clarity improves reliability efforts.

Deliverables:

- Record motivations, outcomes, and business justification in the [strategy and plan template ↗](#).
- Ensure the management baseline provides operational support that aligns to the strategic direction of cloud adoption.

Guidance to support deliverable completion:

- **Understand motivations:** Critical business events and some migration motivations tend to be cost sensitive. These areas can increase the importance of cost control for all later efforts. Other forward-looking motivations related to innovation or growth through migration might be focused more on top-line revenue. Understanding motivations helps you prioritize your cost management.
- **Business outcomes:** Some fiscal outcomes tend to be extremely cost sensitive. When the desired outcomes map to fiscal metrics, you should invest early in the Cost Management governance discipline.
- **Business justification:** The business justification serves as a high-level view of the overall financial plan for cloud adoption. It can be a good source for initial budgeting efforts.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud strategy team	<ul style="list-style-type: none">• Cloud governance team• Cloud operations team• Cloud center of excellence or central IT team

Step 4: Develop a cloud adoption plan

The digital estate (or analysis of the existing IT portfolio) can help you to validate the business justification. It can provide a refined view of the overall IT portfolio. The adoption plan provides clarity on the timeline of activities during adoption.

When you align the adoption plan with the digital estate analysis, you can plan for future operations management dependencies. Understanding the adoption plan also invites the cloud operations team into the development cycles. They can evaluate and plan for any changes to the management baseline that are required to provide workload operations.

Deliverables:

- Update the [strategy and plan template](#) to reflect changes that are needed to achieve the desired strategy. The changes recorded can include:
 - An assessment of the existing digital estate.
 - A cloud adoption plan that reflects the required changes and the work involved.
 - The organizational change that's required to deliver on the plan.
 - A plan for addressing the skills that are needed to enable the existing team to successfully complete the required work.
- Work with the governance team to align cost models and forecast models. This process includes efforts to start optimizing spend through quantitative analysis.

Guidance to support deliverable completion:

- [Gather inventory](#): Establish a source of data for analysis of the digital estate prior to adoption.
- [Best practice: Azure Migrate](#): Use Azure Migrate to gather inventory.
- [Incremental rationalization](#): During incremental rationalization, a quantitative analysis can identify cloud candidates for budgeting purposes.
- [Align cost models and forecast models](#): Use Azure Cost Management + Billing to align cost and forecast models by [creating budgets](#).
- [Build your cloud adoption plan](#): Build a plan with actionable workload, assets, and timeline details.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud strategy team 	<ul style="list-style-type: none"> • Cloud adoption team • Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 5: Implement landing zone best practices

The Ready methodology of the Cloud Adoption Framework focuses heavily on the development of landing zones to host workloads in the cloud. During landing zone implementation, multiple decisions could affect operations. Consult the cloud operations team to help review the landing zone for operations improvements. Also consult the cloud governance team to understand Resource Consistency policies and design guidance that might affect the landing zone design.

Deliverables:

- Deploy one or more landing zones capable of hosting workloads in the short-term adoption plan.
- Ensure that all landing zones meet operations decisions and resource consistency requirements.

Guidance to support deliverable completion:

- [Improve landing zone operations](#): Best practices for improving operations within a given landing zone.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud adoption team 	<ul style="list-style-type: none"> • Cloud operations team • Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 6: Complete waves of adoption effort and change

Long-term operations can be affected by the decisions made during migration and innovation efforts. Maintaining consistent alignment early in adoption processes helps to remove barriers to production releases. It also reduces the effort that's required to introduce new solutions into operations management practices.

Deliverables:

- Test operational readiness of production deployments by using Resource Consistency policies.
- Validate adherence to resource consistency design guidance and operations requirements.
- Document any advanced operations requirements in the [operations management workbook](#).

Guidance to support deliverable completion:

- [Environmental readiness checklist](#)
- [Pre-promotion checklist](#)
- [Production release checklist](#)

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud adoption team	<ul style="list-style-type: none">• Cloud strategy team• Cloud operations team• Cloud governance team• Cloud center of excellence or central IT team

Value statement

These steps help you to implement the controls and processes that are needed to ensure reliability across the enterprise and all hosted resources.

Get started: Ensure consistent performance across a portfolio

Article • 02/28/2023

How do you ensure adequate performance across a portfolio of workloads? The steps in this guide can help you establish processes for maintaining that level of performance.

Performance also depends on other roles and functions. This article maps those supporting functions to help you create alignment among the involved teams.

Centralized operations management is the most common approach to consistent performance across the portfolio. Decisions about operational practices define the operations baseline and any holistic enhancements.

The first step in this guide helps the operations team get started. The subsequent steps help the entire enterprise get started on a shared journey toward enterprise performance across the portfolio of workloads.

Step 1: Establish operations management requirements

The operations management baseline, outlined in the Microsoft Cloud Adoption Framework for Azure, provides a set of controls and cloud-native operations tools to ensure consistent operations. Expanding that baseline with automation tooling provides performance monitoring and automation to meet consistent performance requirements across the portfolio.

Deliverables:

- Enhance the management baseline to include automated remediation tasks related to deviations from performance expectations.
- When workload-specific data patterns or architecture changes are needed to meet performance requirements, use workload-specific operations to provide greater performance controls.
- Document operational decisions across the IT portfolio in the [operations management workbook](#). Focus on including performance automation decisions in the `Operational Compliance` section of the `Baseline` tab.

Guidance to support deliverable completion:

- The [enhanced management baseline](#) article outlines examples of using tools like Azure Automation to add performance-related enhancements. This approach can aid in maintaining consistent performance through basic modifications to the size and scale of supporting assets.
- [Workload-specific operations](#) uses the Microsoft Azure Well-Architected Review to provide guidance on automation for a specific workload. This approach to performance management is particularly useful when workload-specific data should drive operational actions.
- The preceding guidance assumes that an existing implementation of a [management baseline](#) supports the full portfolio of workloads.

 **Note**

Various decisions throughout the cloud adoption lifecycle can have a direct impact on performance. The following steps help outline the partnerships and supporting efforts required to deliver performance across the IT portfolio.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud operations team 	<ul style="list-style-type: none"> • Cloud strategy team • Cloud adoption team • Cloud governance team • Cloud center of excellence or central IT team

Step 2: Consistent application of the management baseline

As the management baseline is improved, it's important to ensure that those improvements carry through to the Resource Consistency governance discipline. Doing so ensures the application of the enhanced baseline in all managed environments.

Deliverable:

Ensure proper application of the enhanced management baseline for all affected systems.

Guidance to support deliverable completion:

- Ensure that all workloads and resources follow [proper naming and tagging conventions](#). Enforce tagging conventions by using [Azure Policy](#), with a specific

emphasis on tags for **criticality**.

- If you're new to cloud governance, establish [governance policies, processes, and disciplines](#) by using the Govern methodology.
- If you're new to the Cost Management discipline, consider following the [article about Cost Management discipline improvements](#), with a focus on the [Implementation](#) section.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud governance team	<ul style="list-style-type: none">• Cloud strategy team• Cloud operations team• Cloud center of excellence or central IT team

Step 3: Define strategy

Strategic decisions directly affect performance, rippling through the adoption lifecycle and into long-term operations. Strategic clarity improves performance efforts across the portfolio. That clarity also helps the operations team understand which workloads need a degree of workload specialization and advanced operations.

Deliverables:

- Record motivations, outcomes, and business justification in the [strategy and plan template](#).
- Ensure that the management baseline provides operational support that aligns with the strategic direction of cloud adoption.

Guidance to support deliverable completion:

- **Understand motivations:** Critical business events and some migration motivations tend to be cost sensitive, which increases the importance of cost control for all later efforts. Other forward-looking motivations related to innovation or growth through migration might be focused more on top-line revenue. Understanding motivations can help you decide how high to prioritize your cost management.
- **Business outcomes:** Some fiscal outcomes tend to be extremely cost sensitive. When the desired outcomes map to fiscal metrics, you should invest in the Cost Management governance discipline early.
- **Business justification:** The business justification serves as a high-level view of the financial plan for cloud adoption. This can be a good source for initial budgeting efforts.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> • Cloud strategy team 	<ul style="list-style-type: none"> • Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 4: Assess and plan for workload adoption

The digital estate (or analysis of the existing IT portfolio) can aid in validating the business justification and provide a refined view of the IT portfolio. The adoption plan provides clarity on the timeline of activities during adoption. Aligning that plan and digital estate analysis provides a means of planning for future dependencies on operations management.

Understanding the plan also invites the cloud operations team into the development cycle. The team can then evaluate and plan for any changes to the management baseline that are required to provide workload operations.

Deliverables:

- Update the [strategy and plan template](#) to reflect changes triggered by the digital estate analysis.
- Work with the cloud operations team to clearly define the criticality and business impact of each workload in the near-term and long-term adoption plan.
- Work with the cloud operations team to establish a timeline for operations readiness.

Guidance to support deliverable completion:

- **Gather inventory:** Establish a source of data for analysis of the digital estate before adoption.
- **Best practice: Azure Migrate:** Use Azure Migrate to gather inventory.
- **Incremental rationalization:** During incremental rationalization, use a quantitative analysis to identify cloud candidates for budgeting purposes.
- **Align cost models and forecast models:** Use Azure Cost Management + Billing to align cost and forecast models by [creating budgets](#).
- **Build your cloud adoption plan:** Build a plan with actionable workload, asset, and timeline details.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> Cloud strategy team 	<ul style="list-style-type: none"> Cloud governance team Cloud operations team Cloud center of excellence or central IT team

Step 5: Expand the landing zones

The Ready methodology of the Cloud Adoption Framework focuses heavily on the development of landing zones to host workloads in the cloud. During landing zone implementation, various decisions can affect operations.

Consult the cloud operations team to help review the landing zone for operations improvements. Also consult the cloud governance team to understand Resource Consistency policies and design guidance, which can affect the landing zone design.

Deliverables:

- Deploy one or more landing zones that can host workloads in the short-term adoption plan.
- Ensure that all landing zones meet operations decisions and resource consistency requirements.

Guidance to support deliverable completion:

- [Improve landing zone operations](#): Best practices for improving operations within a landing zone.

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none"> Cloud adoption team 	<ul style="list-style-type: none"> Cloud operations team Cloud strategy team Cloud governance team Cloud center of excellence or central IT team

Step 6: Adoption

Long-term operations might be affected by the decisions that you make during migration and innovation efforts. Maintaining consistent alignment early in adoption processes helps remove barriers to production release. It also reduces the effort required to onboard new solutions into operations management practices.

Deliverables:

- Test operational readiness of production deployments by using Resource Consistency policies.
- Validate adherence to design guidance for resource consistency and to operations requirements.
- Document any advanced operations requirements in the [operations management workbook](#) ↗.

Guidance to support deliverable completion:

- [Environmental readiness checklist](#)
- [Pre-promotion checklist](#)
- [Production release checklist](#)

Accountable team	Responsible and supporting teams
<ul style="list-style-type: none">• Cloud adoption team	<ul style="list-style-type: none">• Cloud strategy team• Cloud operations team• Cloud governance team• Cloud center of excellence or central IT team

Value statement

The preceding steps will help you implement controls and processes to ensure performance across the enterprise and all hosted resources.

Cloud adoption antipatterns

Article • 04/10/2024

Customers often experience cloud adoption [antipatterns](#). These missteps commonly occur in design, planning, or implementation when migrating to the cloud. Antipatterns can block innovation and prevent businesses from adopting and realizing goals.

The following table lists antipatterns and the methodologies, or cloud adoption phases, that these patterns occur in. The linked articles provide examples of each antipattern and solutions.

[+] Expand table

Methodology	Antipattern	Reference
Strategy	Inadequate motivation	Antipattern: Adopt the cloud without establishing goals
Strategy	Misaligned motivation	Antipattern: Fail to communicate motivations
Plan	Wrong cloud operating model	Antipattern: Choose the wrong cloud operating model
Plan	Wrong service model	Antipattern: Choose the wrong service model
Plan	Replacement instead of modernization	Antipattern: Replace architecture
Ready	Preview services in production	Antipattern: Assume released services are ready for production
Ready	Inaccurate resiliency and availability assumptions	Antipattern: Assume increased resiliency and availability
Ready	IT as cloud provider	Antipattern: Become a cloud provider
Manage	Neglect of business outcomes	Antipattern: Focus on tooling, not business outcomes
Govern	Misaligned shared responsibilities	Antipattern: Misunderstand shared responsibilities
Govern	Inaccurate out-of-the-box security assumptions	Antipattern: Assume out-of-the-box solutions provide security
Govern	Custom compliance or governance frameworks	Antipattern: Use a custom compliance or governance framework

Methodology	Antipattern	Reference
Organize	IT cost centers	Antipattern: Treat IT as a cost center
Organize	Platform development without business approval	Antipattern: Invest in new technology without involving the business
Organize	Core business function outsourcing	Antipattern: Outsource core business functions
Organize	Technical decision makers instead of cloud engineers	Antipattern: Hire technical decision makers instead of developing cloud engineers

Next steps

- [Cloud strategy antipatterns](#)
- [Cloud adoption plan antipatterns](#)

Feedback

Was this page helpful?

 Yes

 No

Develop a cloud adoption strategy

Article • 02/28/2023

The cloud delivers fundamental technology benefits that can help your enterprise execute multiple business strategies. By using cloud-based approaches, you can:

- Improve business agility
- Reduce costs
- Accelerate time to market
- Enable expansion into new markets

To take advantage of this great potential, start by documenting your business strategy in a way that's both understandable to cloud technicians and palatable to your business stakeholders.

The following steps help you document your business strategy efficiently. This approach helps you drive adoption efforts that capture targeted business value in a cross-functional model. Then, you can map your cloud adoption strategy to specific cloud capabilities. You can also map business strategies to reach your expected state of transformation.

<p>1 Define and document your motivations: Meet with key stakeholders and executives to document the motivations behind cloud adoption.</p>
<p>2 Document business outcomes: Engage motivated stakeholders and executives to document specific business outcomes.</p>
<p>3 Evaluate financial considerations: Learn how to use the cloud to make your IT cost structure more flexible. Then, build a business case to adopt the cloud.</p>
<p>4 Understand technical considerations: Discover the technical flexibility, efficiencies, and capabilities that help you build a business case to adopt the cloud.</p>

Take the [Cloud Adoption Strategy Evaluator](#) assessment that will help you build a business case to enable your cloud journey.

Use the results from the assessment—and continue to build your cloud adoption strategy by using the [strategy and plan template](#) ↗ to track the output of each of the

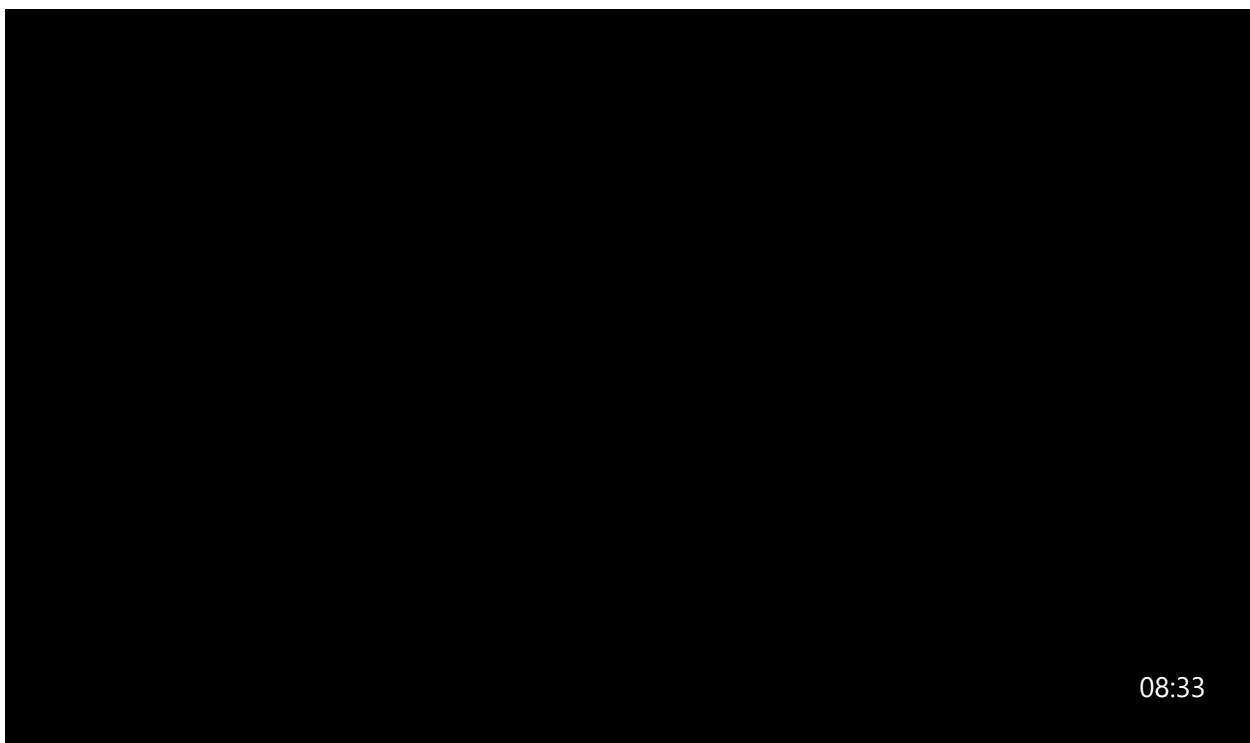
four steps above, documenting your business strategy.

Assess your cloud adoption strategy

Article • 02/28/2023

The Cloud Adoption Framework provides the Cloud Adoption Strategy Evaluator to help you assess your overall cloud adoption strategy. This assessment offers recommendations based on the cloud economics principles that can help your organization create a robust business case and enable a successful cloud adoption.

Watch the following video to learn how the Cloud Adoption Strategy Evaluator can help your cloud adoption efforts.



Cloud Adoption Strategy Evaluator

Every cloud adoption journey is unique, and there's often room for optimization as you start to migrate your organization's services to the cloud. No matter what your situation is, it can be a challenge to build solutions the right way with minimum trial and error.

Several cloud services and different [on-ramps to cloud adoption](#) are available to help you achieve this. Depending on which solution you're considering for your cloud adoption plan, there might be a dozen or more viable paths you can use to start building it. [Microsoft Assessments](#) make it easier to find tailored guidance to assist your organization in building or hosting your workloads in Azure with relevant SLAs. The assessments help you comprehensively assess the current state of your cloud strategy and close any gaps in your skill sets, stakeholder support, business KPIs and budget targets to ensure you have a successful cloud adoption journey.

The Cloud Adoption Strategy Evaluator assesses your strategy posture across distinct areas of the Strategy methodology:

- Identifying motivations
- Documenting expected business outcomes
- Evaluating financial considerations
- Technical considerations in creating a business case

Based on your responses to the assessment questions, we guide you in detail through the categories that are most relevant to your organization. We provide you with a personalized aggregate strategy score that's calculated and averaged across your uniquely identified strategy areas.

As you create your cloud adoption plan and document your strategy for stakeholder review, you'll also receive curated guidance that points to specific tools and templates, plus recommendations based on cloud economics and organizational alignment principles that provide a unified approach you can use to build your business case.

[Take the Cloud Adoption Strategy Evaluator assessment.](#)

Next steps

Understanding the motivations behind your organization's cloud migration can help you achieve more successful business outcomes. The following exercise helps you have a conversation about available options and, ultimately, create positive business outcomes.

[Overview of motivations](#)

Why are we moving to the cloud?

Article • 06/15/2023

"Why are we moving to the cloud?" This is a common question for both businesses and technical stakeholders. If the answer is, "Our board (or CIO or C-level executives) told us to move to the cloud," it might be more difficult for those businesses to achieve their expected outcomes.

This article discusses some of the motivations behind cloud migration that can help produce more successful business outcomes. Understanding these motivations will help you create a conversation about the available options and, ultimately, create positive business outcomes.

Motivations

Various motivations can drive business transformations that are supported by cloud adoption. Several motivations likely apply at the same time. The goal of the list in the following table is to help generate ideas about which motivations are relevant. From there, you can evaluate and assess the potential impacts of the applicable motivations. Your cloud adoption team should meet with the stakeholders, executives, and business leaders and discuss which motivations can help your business's cloud adoption.

Critical business events	Migration	Innovation

Critical business events	Migration	Innovation
Datacenter exit	Cost savings	Preparation for new technical capabilities
Merger, acquisition, or divestiture	Reduction in vendor or technical complexity	Building new technical capabilities
Reduction in capital expenses	Optimization of internal operations	Scaling to meet market demands
End of support for mission-critical technologies	Increase in business agility	Scaling to meet geographic demands
Response to regulatory compliance changes	Preparation for new technical capabilities	Improved customer experiences and engagements
New data sovereignty requirements	Scaling to meet market demands	Transformation of products or services
Reduction of disruptions and improvement of IT stability	Scaling to meet geographic demands	Market disruption with new products or services
Report and manage the environmental impact of your business	Integration of a complex IT portfolio	Democratization and/or self-service environments

Classify your motivations

Your motivations for cloud adoption will likely fall into multiple categories. As you're building the list of motivations, trends will likely emerge. Motivations tend to be associated more with one classification than with others. Use the predominant classification to help guide the development of your cloud adoption strategy.

When responding to critical business events is the highest priority, it's important to [get started with migration](#) early, often in parallel with strategy and planning efforts. Taking this approach requires a growth mindset and a willingness to improve processes based on lessons learned iteratively.

When migration is the highest priority, strategy and planning will play a vital role early in the process. We recommend that you implement the first workload in parallel with planning efforts to help the team understand and anticipate any learning curves associated with cloud adoption.

When innovation is the highest priority, strategy and planning require more investments early in the process. This ensures balance in the portfolio and wise alignment of the investments made during cloud adoption. For more information and guidance, see [Understand the innovation journey](#).

To ensure better decision-making, all participants in the migration process should have a clear awareness of their motivations. The following section outlines how customers can guide and affect decisions through consistent and strategic methodologies.

Motivation-driven strategies and business outcomes

This section highlights the *migration* and *innovation* motivations and their corresponding strategies.

Migration

The *migration* motivations listed near the top of the motivations table are the most common reasons for adopting the cloud but not necessarily the most significant. These outcomes are crucial to achieve, but they're most effectively used to transition to other, more useful worldviews. This essential first step to cloud adoption is often called a *cloud migration*. The [Migrate methodology](#) of the Cloud Adoption Framework outlines the *strategy for executing a cloud migration*.

Some motivations align well with a migration strategy. Motivations at the top of this list can have less business impact than the ones towards the bottom. **Strategies with migration driving motivations** have helped organizations to successfully create business outcomes that:

- Move towards their [Sustainability goals](#)
- Increase cost savings. Read the [customer story ↗](#).
- Reduce vendor or technical complexity.
- Optimize internal operations.
- Increase business agility. Read the [customer story ↗](#).
- Prepare for new technical capabilities.
- Scale to market demand.
- Scale to geographic demand. Read the [customer story ↗](#).

Innovation

Data is the new commodity, and modern applications are the supply chain driving that data into various experiences. In today's business market, it's hard to find a transformative product or service that isn't built on top of data, insights, and customer experiences. The [Innovate methodology](#) of the Cloud Adoption Framework includes *motivations aligned to a technology strategy* that appear lower in the *Innovation* column of the motivation list above.

The motivations below help IT organizations to focus more on innovation than a migration strategy. **Strategies with innovation driving motivations** have helped organizations to successfully create business outcomes that:

- Move towards their [Sustainability goals](#)
- Increase business agility.
- Prepare for new technical capabilities.
- Build new technical capabilities.
- Scale to market demand.
- Scale to geographic demand.
- Improve customer experience and engagement. Read the [customer story ↗](#).
- Transform products or services.

Next steps

Understanding your projected business outcomes, will help facilitate conversations that you'll need to have. These conversations will prove invaluable in documenting your motivations and supporting metrics, in alignment with your business strategy. Next, read an overview of business outcomes that are commonly associated with a move to the cloud.

[Overview of business outcomes](#)

What business outcomes are associated with transformation journeys?

Article • 02/28/2023

The most successful transformation journeys start with a business outcome in mind. Cloud adoption can be a costly and time-consuming effort. Fostering the right level of support from IT and other areas of the business is crucial to success. This article series is designed to help customers identify business outcomes that are concise, defined, and drive observable results or change in business performance, supported by a specific measure.

During any cloud transformation, the ability to speak in terms of business outcomes supports transparency and cross-functional partnerships. The business outcome framework starts with a simple template to help technically minded individuals document and gain consensus. This template can be used with several business stakeholders to collect a variety of business outcomes, which could each be influenced by a company's transformation journey. Feel free to use this template electronically or, better still, draw it on a whiteboard to engage business leaders and stakeholders in outcome-focused discussions.

To learn more about business outcomes and the business outcome template, see [Documenting business outcomes](#), or download the [business outcome template](#).

Optimize your cloud investment using cloud economics

What are the fundamentals of cloud economics in Azure? Whether you're running existing workloads or designing new solutions in Azure, learn best practices guidance to navigate the economics of the cloud for your organization, and optimize your operating costs in Azure based on your specific workloads. Get started with successfully building your cloud business case with key financial and technical guidance, and maximize the full benefit of your cloud investment.

Learn more about [how cloud economics works](#).

Prepare for conversations with different personas

The following are a few business outcomes that tend to trigger conversations with various personas:

- **Finance leadership:** Increase profitability while driving compliance.
- **Marketing:** Acquire and retain customers, and build a reputation.
- **Sales:** Accelerate sales and improve customer lifetime value.
- **Human resources:** Retain, recruit and empower employees.
- **Executive leadership:** Meeting market growth requirements and environmental sustainability metrics.
- **Sustainability leads:** Driving sustainability as a strategy throughout the business, aligned with the company's commitments.

Sample outcomes by category

Speaking in business outcomes can feel like a foreign language to many technically minded individuals. To help ease translation, we curate a set of business outcome examples. You can use the following examples to inspire and demonstrate business outcomes that are based on actual transformation journeys.

To help you find business outcomes more easily, we've separated them into the following categories. This approach tends to drive consensus-building conversations across business units.

Fiscal outcomes

Financial or fiscal performance is the cleanest business outcome for many business leaders, but not the only one.

View samples of [fiscal outcomes](#).

Agility outcomes

Today's fast-changing business environment places a premium on time. The ability to respond to and drive market change quickly is the fundamental measure of business agility.

View samples of [agility outcomes](#).

Reach outcomes

In a constantly shrinking market, global reach (ability to support global customers and users) can be measured by compliance in geographies that are relevant to the business.

View outcomes related to [global reach](#).

Customer engagement outcomes

Social marketplaces are redefining winners and losers at an unheard-of pace. Responding to user needs is a key measure of customer engagement.

Learn more about [customer engagement outcomes](#).

Performance outcomes

Performance and reliability are assumed. When either falters, reputation damage can be painful and long-lasting.

Learn more about [performance outcomes](#).

Sustainability outcomes

Organizations are increasingly discussing environmental goals and sustainability targets. Environmental sustainability is quickly moving from a "nice to have" function of Corporate Social Responsibility (CSR) to a centerpiece of business strategy, increasingly embedded into everything a company does.

Learn more about [sustainability goals](#).

Each business outcome listed in the preceding categories can help facilitate a focused conversation among your business and technical team members. However, you shouldn't limit your conversations to these generic samples. Understanding the unique needs of your business and building outcomes that match maximizes the value of a cloud transformation.

Next steps

Learn more about [fiscal outcomes](#).

[Fiscal outcomes](#)

Data innovations

Article • 04/21/2023

Many companies want to migrate their existing data warehouse to the cloud. They may be motivated by a number of factors, including:

- No hardware to buy or maintenance costs.
- No infrastructure to manage.
- The ability to switch to a secure, scalable, and low-cost cloud solution.

For example, Azure Synapse Analytics is a cloud-native, pay-as-you-go service which provides an analytical database management system for organizations. Azure technologies can help modernize your data warehouse after migration, and extend your analytical capabilities to drive new business value.

A data warehouse migration project involves many components. These include schema, data, extract-transform-load (ETL) pipelines, authorization privileges, users, BI tool semantic access layers, and analytic applications.

After your data warehouse has been migrated to Azure Synapse Analytics, you can take advantage of other technologies in the Microsoft analytical ecosystem. Doing so allows you to not only modernize your data warehouse but also bring together insights produced in other analytical data stores in Azure.

You can broaden ETL processing to ingest data of any type into Azure Data Lake Storage, and you can prepare and integrate it at scale by using Azure Data Factory. This produces trusted, commonly understood data assets that can be consumed by your data warehouse, and also be accessed by data scientists and other applications. You can build real-time, batch-oriented analytical pipelines. You can also create machine learning models that can deploy to run in batch, in real-time on streaming data, and on-demand.

In addition, you can use PolyBase to go beyond your data warehouse, simplifying access to insights produced in multiple underlying analytical platforms on Azure. You create holistic, integrated views in a logical data warehouse to gain access to streaming, big data, and traditional data warehouse insights from BI tools and applications.

Many companies have had data warehouses running in their datacenters for years, to enable users to produce business intelligence. Data warehouses extract data from known transaction systems, stage the data, and then clean, transform, and integrate it to populate data warehouses.

Use cases, business cases, and technology advances all contribute to how Azure Synapse Analytics can help you with data warehouse migration. The following sections list many

of these examples.

Use cases

- Connected product innovation
- Factory of the future
- Clinical analytics
- Compliance analytics
- Cost-based analytics
- Omnichannel optimization
- Personalization
- Intelligent supply chain
- Dynamic pricing
- Procurement analytics
- Digital control tower
- Risk management
- Customer analytics
- Fraud detection
- Claims analytics

Business cases

- Build end-to-end analytics solutions with a single analytics service.
- Use the Azure Synapse Analytics studio, which provides a unified workspace for data prep, cloud-scale analytics, data warehousing, big data, and AI tasks.
- Build and manage pipeline with a no-code visual environment, automate query optimization, build proofs of concept, and use Power BI, all from the same analytics service.
- Deliver your data insights to data warehouses and big data analytics systems.
- For mission-critical workloads, optimize the performance of all queries with intelligent workload management, workload isolation, and limitless concurrency.
- Edit and build Power BI dashboards directly from Azure Synapse Analytics.
- Reduce project development time for BI and machine learning projects.
- Easily share data with just a few clicks by using Azure Data Share integration within Azure Synapse Analytics.
- Implement fine-grained access control with column-level security and native row-level security.
- Automatically protect sensitive data in real time with dynamic data masking.
- Industry-leading security with built-in security features like automated threat detection and always-on data encryption.

Technology advances

- No hardware to buy or maintenance costs so you pay only for what you use.
- No infrastructure to manage, so you can focus on competitive insights.
- Massively parallel SQL query processing with dynamic scalability when you need it, and the option to shut down or pause when you don't.
- Ability to independently scale storage from compute.
- You can avoid unnecessary, expensive upgrades caused by the staging areas on your data warehouse getting too big, taking up storage capacity, and forcing an upgrade. For example, move the staging area to Azure Data Lake Storage. Then process it with an ETL tool like Azure Data Factory or your existing ETL tool running on Azure at lower cost.
- Avoid expensive hardware upgrades by processing ETL workloads in Azure, by using Azure Data Lake Storage and Azure Data Factory. This is often a better solution than running on your existing data warehouse DBMS with SQL query processing doing the work. As staging data volumes increase, more storage and compute power underpinning your on-premises data warehouse is consumed by ETL. This in turn affects the performance of query, reporting, and analysis workloads.
- Avoid building expensive data marts that use storage and databases software licenses on on-premises hardware. You can build them in Azure Synapse Analytics instead. This is especially helpful if your data warehouse is a data vault design, which often causes an increased demand for data marts.
- Avoid the cost of analyzing and storing high-velocity, high-volume data on on-premises hardware. For example, if you need to analyze real-time, machine generated data like click-stream and streaming IoT data in your data warehouse, you can use Azure Synapse Analytics.
- You can avoid paying a premium for storing data on expensive warehouse hardware in the datacenter as your data warehouse grows. Azure Synapse Analytics can store your data in cloud storage at a lower cost.

Next steps

[Data democratization](#)

Support business with data innovation

Article • 02/28/2023

Many companies keep data warehouses in their datacenters to help different parts of their business analyze data and make decisions. Sales, marketing, and finance departments rely heavily on these systems in order to produce standard reports and dashboards. Companies also employ business analysts to perform ad hoc querying and analysis of data in data marts. These data marts use self-service business intelligence tools to perform multidimensional analysis.

A business that's supported by data innovation and a modern data estate can empower a broad range of contributors, from an IT stakeholder to a data professional and beyond. They can take action on this repository of centralized data, which is often referred to as "the single source of truth."

Azure Synapse Analytics is a single service for seamless collaboration and accelerated time-to-insight. To understand this service in more detail, first consider the various roles and skills involved in a typical data estate:

Data warehousing: Database admins support the management of data lakes and data warehouses while intelligently optimizing workloads and automatically securing data.

Data integration: Data engineers use a code-free environment to easily connect multiple sources and types of data.

Big data and machine learning: Data scientists build proofs of concept rapidly and provision resources while working in the language of their choice (for example, T-SQL, Python, Scala, .NET, or Spark SQL).

Management and security: IT pros protect and manage data more efficiently, enforce privacy requirements, and secure access to cloud and hybrid configurations.

Business intelligence: Business analysts securely access datasets, build dashboards, and share data within and outside their organization.

An overview of classic data warehouse architecture

The following diagram shows an example of a classic data warehouse architecture.

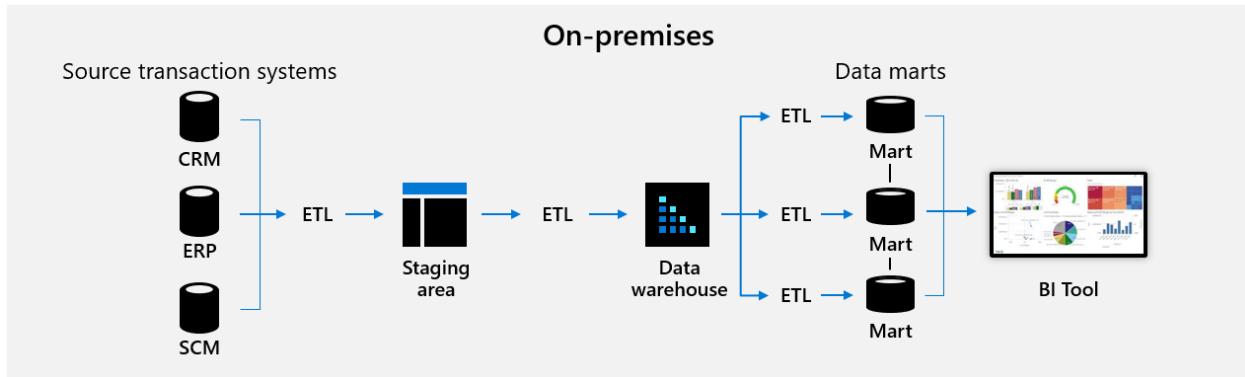


Figure 1: Classic data warehouse architecture.

Known structured data is extracted from core transaction processing systems and copied into a staging area. From there, it's cleaned, transformed, and integrated into production tables in a data warehouse. It's common for several years of historical transaction data to build up here. This provides the data needed to understand changes in sales, customer purchasing behavior, and customer segmentation over time. It also provides yearly financial reporting and analysis to help with decision making.

From there, subsets of data are extracted into data marts to analyze activity associated with a specific business process. This supports decision making in a specific part of the business.

For a business to run efficiently, it needs all types of data for the different skills and roles described earlier. You need raw data that has been cleansed for data scientists to build machine-learning models. You need clean and structured data for a data warehouse to provide reliable performance to business applications and dashboards. Most importantly, you need to be able to go from raw data to insights in minutes, not days.

Azure Synapse Analytics has a native, built-in business intelligence tool with Microsoft Power BI. Here, one service within one interface supports you to quickly transform raw data into a dashboard that displays insights.

Next steps

[Data innovations](#)

Examples of fiscal outcomes

Article • 02/28/2023

The three basic concepts of fiscal conversation are:

- **Revenue:** Will more money come into the business as a result of the sales of goods or services?
- **Cost:** Will less money be spent in the creation, marketing, sales, or delivery of goods or services?
- **Profit:** Although they're rare, some transformations can both increase revenue and decrease costs. This is a profit outcome.

This article explains these fiscal outcomes within the context of a cloud transformation.

[Sentara Healthcare](#) moved its data to Azure to provide:

- better mobile app experiences for members
- faster access to critical diagnostic data for clinicians
- reduced IT team costs
- improved patient care
- a more agile, change-ready organization overall

ⓘ Note

The following examples are hypothetical and should not be considered a guarantee of returns when adopting any cloud strategy.

Revenue outcomes

New revenue streams

The cloud can create opportunities that allow you to deliver new products to customers or deliver existing products in a new way. New revenue streams are innovative, entrepreneurial, and exciting for people in the business world.

In some companies, new revenue streams are prone to failure and considered high risk. When a revenue-related outcome is proposed by IT in one of these companies, resistance is likely. Partnering with a business leader who's a proven innovator lends credibility to proposed outcomes, and demonstrating revenue stream validation early in the process can help prevent roadblocks.

- **Example:** A company has been selling books for over a hundred years. An employee of the company realizes that the content can be delivered electronically. The employee creates a device that's sold in the bookstore, that allows the same books to be downloaded directly, driving \$x in new book sales.

Revenue increases

Through its global scale and digital reach the cloud can help businesses increase revenues from existing revenue streams. This type of outcome often results from an alignment with sales or marketing leadership.

- **Example:** A company sells widgets. It could sell more widgets, if its salespeople could securely access the company's digital catalog and stock levels. Unfortunately, that data is only in the company's ERP system, which can only be accessed through network-connected devices. If the company creates a service façade to interface with the ERP and expose the catalog list and non-sensitive stock levels to an application in the cloud, salespeople can access the data they need while onsite with a customer. Extending on-premises Active Directory using Azure Active Directory (Azure AD) and integrating role-based access into the application allows the company to ensure their data stays safe. This simple change can affect revenue from an existing product line by x%.

Profit increases

Seldom does a single effort simultaneously increase revenue and decrease costs. However, when it does, align the outcome statements from one or more of the revenue outcomes with one or more of the cost outcomes to communicate the desired outcome.

Cost outcomes

Cost reduction

Cloud computing can reduce capital expenses for hardware and software, datacenter setup, on-site datacenter operation, and more. The cost of servers, round-the-clock electricity for power and cooling, and IT experts to manage infrastructure adds up fast. Shutting down a datacenter can reduce your capital expense commitments. This is commonly referred to as "getting out of the datacenter business." The resulting cost reduction is typically measured in dollars in the current budget, which can span one to five years depending on how your CFO manages finances.

- **Example #1:** A company's datacenter consumes a large percentage of their annual IT budget. IT chooses to conduct a cloud migration and transitions that datacenter's assets to infrastructure as a service (IaaS) solutions, creating a three-year cost reduction.
- **Example #2:** A holding company recently acquired a new company. The terms of the acquisition dictate that the new entity should be removed from its current datacenters within six months. Failure to do so will result in a fine of \$1 million USD per month to the holding company. Moving the new entity's digital assets to the cloud in a cloud migration can allow a quick decommission of the old assets.
- **Example #3:** An income tax company catering to consumers experiences 70 percent of its annual revenue during the first three months of the year. For the remainder of the year, its large IT investment is relatively dormant. A cloud migration can allow IT to deploy the compute/hosting capacity required for those three months. During the remaining nine months, they can reduce the compute footprint and IaaS costs.

Example: Coverdell

Coverdell modernizes their infrastructure to drive record cost savings with Azure. Coverdell's decision to invest in Azure, and to unite their network of websites, applications, data, and infrastructure within this environment, led to more cost savings than the company expected. Their migration to an Azure-only environment eliminated \$54,000 USD in monthly costs for co-location services. With their new united infrastructure, Coverdell expects to save an estimated \$1M USD over the next two to three years.

"Having access to the Azure technology stack opens the door for some scalable, easy-to-implement, and highly available solutions that are cost effective. This allows our architects to be much more creative with the solutions they provide."

Ryan Sorensen

Director of Application Development and Enterprise Architecture

Coverdell

Cost avoidance

Terminating a datacenter can also provide cost avoidance by preventing future refresh cycles. A refresh cycle is the process of buying new hardware and software to replace aging on-premises systems. In Azure, hardware and OS are routinely maintained, patched, and refreshed at no additional cost to customers. This allows CFOs to remove

planned future spend from long-term financial forecasts. Cost avoidance is measured in dollars and differs from cost reduction, generally focusing on a future budget that has not been fully approved yet.

- **Example:** A company's datacenter is up for a lease renewal in six months. The datacenter has been in service for eight years. Four years ago, the company spent millions of dollars ensuring all servers were refreshed and virtualized. The company plans to refresh the hardware and software again next year. Migrating that datacenter's assets as part of a cloud migration allows cost avoidance by removing the planned refresh from next year's forecasted budget. It can also produce cost reduction by decreasing or eliminating real estate lease costs.

Capital expenses and operating expenses

The two primary cost options are capital expenses and operating expenses.

The following terms can help you define the differences between capital expenses and operating expenses during business discussions about your transformation journey.

- **Capital** is the money and assets owned by a business to contribute to a particular purpose, such as increasing server capacity or building an application.
- **Capital expenditures** generate benefits over a long period. These expenditures are generally non-recurring and result in the acquisition of permanent assets. Building an application can qualify as a capital expenditure.
- **Operating expenditures** are a business' ongoing costs. Consuming cloud services in a pay-as-you-go model can qualify as an operating expenditure.
- **Assets** are economic resources that can be owned or controlled to produce value. Servers, data lakes, and applications can be considered assets.
- **Depreciation** is a decrease in the value of an asset over time. More relevant to the capital expense versus operating expense conversation, depreciation is how the costs of an asset are allocated across the periods in which they are used. For example, if you build an application this year but it's expected to have an average shelf life of five years (like most commercial applications), the cost of the development team and the tools required to create and deploy the code base would be depreciated evenly over five years.
- **Valuation** is the process of estimating a company's worth. In most industries, valuation is based on a company's ability to generate revenue and profit while respecting the operating costs required to create the goods that provide their revenue. In some industries, like retail, or in some transaction types, like private equity, assets and depreciation can play a large part in a company's valuation.

Various executives, including the chief investment officer (CIO), often debate the best use of capital to grow the company in the desired direction. Giving the CIO a means of converting contentious capital expense conversations into clear accountability for operating expenses can be an attractive outcome by itself. In many industries, chief financial officers (CFOs) actively look for ways to better associate fiscal accountability with the cost of goods being sold.

However, before you associate any transformation journey with this type of capital versus operating expense conversion, it's wise to meet with members of the CFO or CIO teams to see which cost structure your business prefers. In some organizations, reduction of capital expenses in favor of operating expenses is a highly undesirable outcome. As mentioned previously, this approach is sometimes seen in retail, holding, and private equity companies that place higher value on traditional asset accounting models, which place little value on IP. It's also seen in organizations that had negative experiences when outsourcing IT staff or other functions in the past.

The following example describes a situation where an operating expense model is a viable business outcome.

- **Example:** A company's datacenter is currently depreciating at $\$x$ USD per year for the next three years. It is expected to require an additional $\$y$ USD to refresh its hardware next year. We can convert the capital expenses to an operating expense model at an even rate of $\$z$ USD per month, enabling better management and accountability for the operating costs of technology.

Next steps

Learn about [agility outcomes](#).

[Agility outcomes](#)

Examples of agility outcomes

Article • 02/28/2023

As discussed in the [business outcomes overview](#), several potential business outcomes can serve as the foundation of any conversation with the business about its transformation journey. This article focuses on the timeliest business measure: business agility. Understanding your company's market position and competitive landscape can help you articulate business outcomes that are the target of the business's transformation journey.

Traditionally, a company's chief information officer (CIO) and IT team were considered the source of stability for the business's core, mission-critical processes. It's still true. A business usually doesn't function well if its IT platform is unstable. But in today's business world, much more is expected of IT. By partnering with the business, IT can expand beyond a simple cost center to provide market advantages. Many CIOs and executives assume that stability is simply a baseline for IT. For these leaders, business agility is the measure of IT's contribution to the business.

The [Academy of Motion Picture Arts and Sciences](#) is an example of how an organization can meet and exceed its business objectives through cloud migration. The Academy used Azure and Visual Studio to migrate its legacy web applications to the cloud. Through cloud adoption, the Academy was able to innovate, increase efficiencies, and deliver new streaming applications to members in a rich, responsive, and cross-platform experience.

Why is agility so important?

Markets change at a faster pace today than ever before. In a 2016 study that looked at turnover rate in companies' Fortune 500 status, of the 500 companies that were on the Fortune 500 list in 1955, 61 years later, only 57 of those companies were still on the list. The turnover rate of 88.6 percent points to an unprecedented rate of market change. IT agility, or even business agility, is unlikely to affect an organization's listing in the Fortune 500, but these numbers help us understand the pace at which markets continue to change.

Both for upstarts and for established organizations, business agility can be the difference between success or failure in a business initiative. Quickly adapting to market changes can help a business ring-fence existing customers or claim market share from competitors. The agility-related outcomes in the following sections can help you articulate the value of migrating to the cloud in the transformation journey.

Time-to-market outcome

During cloud-enabled innovation efforts, time to market is a key measure of IT's ability to address market change. In many cases, a business leader might have an existing budget to create an application or launch a new product. Clearly communicating a time-to-market benefit can motivate that leader to redirect budget to the IT transformation journey.

Example 1: The European division of a US-based company needs to comply with data privacy regulations by protecting customer data in a database that supports UK operations. The company division's existing version of SQL Server doesn't support the row-level security that's required. An in-place upgrade would be too disruptive. Instead, the company uses Azure SQL Database to replicate and upgrade its customer database. It successfully adds the required compliance measure in a matter of weeks.

Example 2: A logistics company has discovered an untapped segment in the market. The company's largest competitor has made the same discovery. To capture the market share before its competitor, the logistics company needs a new version of its flagship application. By adopting cloud-enabled app innovation and a DevOps-driven development approach, the company embraces its customers' obsession and advances beyond its slower, legacy competitor time to market by months. Gaining time on market entrance secured the customer base for the company.

Example 3: In this example, a healthcare system transformed online services into a friendly digital experience. To transform its digital services, Aurora Health Care migrated its websites to the Microsoft Azure platform and adopted a strategy of continuous innovation.

Jamey Shiels, Vice President of Digital Experience at Aurora Health Care, reports that, "as a team, we're focused on high-quality solutions and speed. Choosing Azure was a very transformative decision for us."

Provision time

Changes in a company's operational demands frequently create a need for new IT services or to scale existing services. It might take a company weeks to acquire and provision new hardware and virtual resources. After cloud migration, a company's IT team can use self-service provisioning to put the required resources in place in hours.

Example: To fulfill the operational demands of its business, a consumer packaged goods company needs to create and tear down hundreds of database clusters each year. On-premises virtual hosts can provision quickly, but the process of recovering virtual assets

is slow and a strain on the team. Because of the time involved in the process, the on-premises environment suffers from wasted resources and seldom can keep up with the demand. After the company migrates its virtual assets to the cloud, its IT team can more easily manage resources through self-provisioning and scripts. The company also now uses a chargeback model for billing. Through cloud migration, the business can move as quickly as it needs to, but it can still be accountable for the cost of the resources it requires. In the cloud model, deployment is limited only by the business's budget.

Next steps

Learn more about [reach outcomes](#).

[Reach outcomes](#)

Examples of global reach outcomes

Article • 02/28/2023

As discussed in [Business outcomes](#), several potential business outcomes can serve as the foundation for any transformation journey conversation with the business. This article focuses on a common business measure: reach. *Reach* is a concise term that, in this case, refers to a company's globalization strategy. Understanding the company's globalization strategy helps you better articulate the business outcomes that are the target of a business's transformation journey.

Fortune 500 and smaller enterprises have focused on the globalization of services and customers for over three decades, and most business are likely to engage in global commerce as this globalization continues to pull focus. Hosting datacenters around the world can consume more than 80 percent of an annual IT budget, and wide-area networks using private lines to connect those datacenters can cost millions of dollars per year. Therefore, supporting global operations is both challenging and costly.

Cloud solutions move the cost of globalization to the cloud provider. In Azure, customers can quickly deploy resources in the same region as customers or operations, without buying and provisioning a datacenter. Microsoft owns one of the largest wide-area networks in the world, connecting datacenters around the globe. Connectivity and global operating capacity are available to global customers on demand.

[Walgreens Boots Alliance](#) moved on-premises applications and IT resources in a heterogeneous Linux and Windows environment to the cloud, benefiting from improved performance and data centralization, and helping the company provide better customer service.

Global access

Expanding into a new market can be one of the most valuable business outcomes during a transformation. The ability to quickly deploy resources in market without a longer-term commitment allows sales and operations leaders to explore options that wouldn't have been considered in the past.

Manufacturing example

A cosmetics manufacturer has identified a trend. Some products are being shipped to the Asia Pacific region even though no sales teams are operating in that region. The minimum systems required by a remote sales force are small, but latency prevents a

remote access solution. To capitalize on this trend, the vice president of sales wants to experiment with sales teams in Japan and South Korea. Because the company has undergone a cloud migration, it was able to deploy the necessary systems in both Japan and South Korea within days. This allowed the vice president of sales to grow revenue in the region within three months. Those two markets continue to outperform other parts of the world, leading to sales operations throughout the region.

Retail example

An online retailer that ships products globally can engage with their customers across time zones and multiple languages. The retailer uses Azure Bot Service and various features in Azure Cognitive Services, such as Translator, Language Understanding (LUIS), QnA Maker, and Text Analytics. This ensures their customers are able to get the information they need when they need it, and that it's provided to them in their language. The retailer uses the [Personalizer service](#) to further customize the experience and catalog offerings for their customers, ensuring geographical tastes, preferences, and availability are reflected.

Data sovereignty

Operating in new markets introduces additional governance constraints. Azure provides compliance offerings that help customers meet compliance obligations across regulated industries and global markets. For more information, see the [overview of Microsoft Azure compliance](#).

Example

A US-based utilities provider was awarded a contract to provide utilities in Canada. Canadian data sovereignty law requires that Canadian data stays in Canada. This company had been working their way through a cloud-enabled application innovation effort for years. As a result, their software was deployed through fully scripted DevOps processes. With a few minor changes to the code base, they were able to deploy a working copy of the code to an Azure datacenter in Canada, meeting data sovereignty compliance and retaining the customer.

Next steps

Learn more about customer engagement outcomes.

[Customer engagement outcomes](#)

Examples of customer engagement outcomes

Article • 02/28/2023

As discussed in the [business outcomes overview](#), several potential business outcomes can serve as the foundation for any transformation journey conversation with the business. This article focuses on a common business measure: customer engagement. Understanding the needs of customers, and the ecosystem around customers, helps you to articulate the business outcomes that are the target of a business's transformation journey.

During cloud-enabled data innovation efforts, you can assume that customers are engaged. The following functions are potentially disruptive and require a high degree of customer engagement:

- Aggregating data
- Testing theories
- Advancing insights
- Informing cultural change

Customer engagement outcomes are about meeting and exceeding customer expectations. As a baseline for customer engagements, customers assume that products and services perform and are reliable. When they're not, it's easy for an executive to understand the business value of performance and reliability outcomes. For more advanced companies, the speed of integrating learnings and observations from this process is a fundamental business outcome.

[Descartes](#) chose Microsoft Azure as its preferred platform, and successfully migrated its Descartes MacroPoint solution to Azure SQL Database to provide greater flexibility for customers, and focus internal resources on extending product value.

Cycle time

During customer-obsessed transformations such as a cloud-enabled application innovation effort, customers respond from direct engagement. They also appreciate seeing their needs met quickly by the development team. Cycle time is a Six Sigma term that refers to the duration from the start to the finish of a function. For business leaders who invest heavily in improving customer engagement, cycle time can be a strong business outcome.

Example:

A services company that provides business-to-business (B2B) services is trying to retain market share in a competitive market. Customers who have left for a competing service provider found that their overly complex technical solution interferes with their business processes, and is the primary reason for leaving. In this case, cycle time is imperative.

It currently takes 12 months for a feature to progress from request to release. If it's prioritized by the executive team, this cycle can shorten from nine to six months. The team can cut cycle time down to one month through a cloud-enabled application innovation effort, cloud-native application models, and Azure DevOps integration. This frees business and application development teams to interact more directly with customers.

Intelligent contact center

Customer satisfaction and experience are at the core of successful organizations. Freeing your employees to focus on superior customer service can strongly affect customer loyalty and retention. With the AI technology available today, many steps during a customer call can be automated, enabling the contact center agent more time to focus on delivering superior customer service.

Example:

An insurance company has implemented digital agents to respond rapidly to customer requests. These digital agents are available through the company website and mobile app, by building an Azure Bot Service solution. Extending an enhanced customer service experience to their contact center, the insurance company implemented live call transcription, sentiment analysis, and key phrase detection. These help the contact center agent with recommended next steps and form processing. This led to reduced repetition from the customer calling the contact center, and enabled the contact center agent to focus more on providing a great customer experience.

Next steps

Learn more about performance outcomes.

[Performance outcomes](#)

Examples of performance outcomes

Article • 06/15/2023

As discussed in [Business outcomes](#), several potential business outcomes can serve as the foundation for any transformation journey conversation with the business. This article focuses on a common business measure: performance.

In today's technological society, customers assume that applications will perform well and always be available. When this expectation isn't met, it causes reputation damage that can be costly and long-lasting.

Performance

The biggest cloud computing services run on a worldwide network of secure datacenters, which are regularly upgraded to the latest generation of fast and efficient computing hardware. This provides several benefits over a single corporate datacenter, such as reduced network latency for applications and greater economies of scale.

Transform your business and reduce costs with an energy-efficient infrastructure that spans more than 100 highly secure facilities worldwide, linked by one of the largest networks on earth. Azure has more global regions than any other cloud provider. This translates into the scale that's required to bring applications closer to users around the world, preserve data residency, and provide comprehensive compliance and resiliency options for customers.

- **Example 1:** A services company worked with a hosting provider that hosted multiple operational infrastructure assets. Those systems suffered from frequent outages and poor performance. The company migrated its assets to Azure to take advantage of the SLA and performance controls of the cloud. Any downtime would cost the company approximately \$15,000 USD per minute of outage. With between four and eight hours of outage per month, it was easy to justify this organizational transformation.
- **Example 2:** A consumer investment company was in the early stages of a cloud-enabled application innovation effort. Agile processes and DevOps were maturing well, but application performance was spiky. As a more mature transformation, the company started a program to monitor and automate sizing based on usage demands. The company eliminated sizing issues by using Azure performance management tools, resulting in a surprising 5 percent increase in transactions.

Reliability

Cloud computing makes data backup, disaster recovery, and business continuity easier and less expensive, because data can be mirrored at multiple redundant sites on the cloud provider's network.

One of IT's crucial functions is ensuring that corporate data is never lost and applications stay available despite server crashes, power outages, or natural disasters. You can keep your data safe and recoverable by backing it up to Azure.

Azure Backup is a simple solution that decreases your infrastructure costs while providing enhanced security mechanisms to protect your data against ransomware. With one solution, you can protect workloads that are running in Azure and on-premises across Linux, Windows, VMware, and Hyper-V. You can ensure business continuity by keeping your applications running in Azure.

Azure Site Recovery makes it simple to test disaster recovery by replicating applications between Azure regions. You can also replicate on-premises VMware and Hyper-V virtual machines and physical servers to Azure to stay available if the primary site goes down. And you can recover workloads to the primary site when it's up and running again.

- **Example:** An oil and gas company used Azure technologies to implement a full site recovery. The company chose not to fully embrace the cloud for day-to-day operations, but the cloud's business continuity and disaster recovery (BCDR) features still protected their datacenter. As a hurricane formed hundreds of miles away, their implementation partner started recovering the site to Azure. Before the hurricane touched down, all mission-critical assets were running in Azure, preventing any downtime.

Next steps

Learn how to use the business outcome template.

[Use the business outcome template](#)

Sustainability outcomes and benefits for business

Article • 06/03/2024

Though the impact and benefits of the cloud have been traditionally measured with financial and efficiency metrics, it's become more common for customers to seek to understand how the cloud can help them to achieve their sustainability and environmental goals. Cloud computing can support your organization to reduce carbon emissions, use resources more efficiently, and lessen your environmental footprint.

Actively working toward an increased sustainability can help increase your revenue, reduce the operating costs, and improve your brand trust. Driving sustainability in your organization have several benefits:

- **Track and control carbon emissions:** Use the [emissions impact dashboard](#) to track carbon emissions. Reveal insights that can justify a data center migration. Use the [Azure Migration and Modernization Program](#) to get expert help needed to set up the cloud environments.
- **Migrate to reduce the carbon footprint:** Migrating to the cloud can be up to [98% more carbon efficient](#). Additionally, use the Azure Well-Architected Framework [Sustainability workload guidance](#) to further enhance your workloads and reduce the environmental footprint.
- **Drive sustainable innovation in the cloud:** Meet customer demands for clean energy solutions and sustainable products.

Microsoft has been leading in many of these areas. The company has been operating as carbon-neutral since 2012 and has made a commitment to be carbon-negative by 2030. [The carbon benefits of cloud computing](#), a study on the Microsoft cloud in partnership with WSP, supports research on how moving on-premises datacenters to the Microsoft cloud can significantly reduce carbon footprints.

[Bühler Group's](#) IoT platform, known as Bühler Insights and powered by Azure IoT Hub, generates vital data so customers can monitor machine performance and generate accurate records for every product batch, helping food producers optimize safety, sustainability, and transparency across the supply chain.

Emerging software design principles

More opportunities have arisen recently to design software that is more sustainable. [Green Software](#) is an emerging discipline with principles, patterns, philosophies,

practices, and competencies to define, develop and run sustainable software applications. The sooner this discipline is adopted into your sustainability journey, the better, as it will help to reduce the carbon emissions consumed by applications.

Read more about building [Sustainable workloads on Azure](#) in the Azure Well-Architected Framework guidance.

The Microsoft sustainability journey

The Microsoft journey started over a decade ago when the company started to apply new business practices and adopt cloud technology. We lowered our carbon emissions by 30 percent in 2009. Since then, Microsoft has made large strides forward by investing in reducing the company's carbon footprint further. Microsoft has focused on these four areas:

- **Carbon:** Cutting energy consumption across corporate offices, charging carbon tax to business divisions, and using cloud-powered technology to lower emissions.
- **Ecosystem:** Making a commitment to green datacenters and purchasing of 1.1 billion kilowatt-hours of green energy.
- **Water:** Reducing use intensity and investing in technology for managing water.
- **Waste:** Practicing responsible sourcing, recycling, and disposal; using software and technology to make buildings more efficient.

Additionally, there are four main drivers contributing to a lowered energy and carbon footprint of the Microsoft Cloud:

- **IT operational efficiency, IT equipment efficiency, and datacenter infrastructure efficiency:** reduce the energy required to deliver the services.
- **Purchase of new renewable electricity:** will power 100 percent of electricity consumed in Microsoft datacenters, buildings, and campuses by 2025.

Read more about how Microsoft's [commitment to a planet-sized challenge](#) ↗ has helped us plan and achieve sustainability goals.

Additionally, read how Microsoft [measures datacenter water and energy use to improve Azure Cloud sustainability](#) ↗ .

Building a sustainability strategy

Sustainability continues to gain importance as a performance indicator for organizations. As the sustainability transformation is making its mark on companies across industries, adding pressure from new types of stakeholders and challenging

existing profit pools while creating opportunities to open new ones, companies are increasingly forced to respond effectively with new types of solutions and tech-enabled approaches.

This transformation is good for business. Research from multiple sources indicates that sustainability front runners have a lower cost-of-capital, deliver superior equity market returns, get easier access to new markets by creating new types of products and services, and/or are better at managing risk and ensuring more resilient operations.

Read about the [Sustainability Executive Playbook](#)

Common considerations for building a sustainability strategy could include:

- Developing new ways of working to increase productivity
- Migrating resources to a more carbon efficient infrastructure
- Net Zero commitments
- Improving emissions recording
- Increased Operational efficiencies
- Improving societal outcomes by co-developing with partners

Build green teams

Initiate the idea of building "green teams" that can have different sustainability metrics depending on the served domain, not dependent on the central sustainability team guidance but contributing to the overall green targets of the company.

Goals and metrics for teams owning sustainability

At Microsoft, we have a dedicated [sustainability science team](#) whose mission is to ensure that our sustainability work is grounded in the best available science. This drives our work in sustainability, from our climate commitments to partnering with our customers and partners on codesigning new solutions.

Establishing goals and metrics for teams owning sustainability in your organization is essential. Metrics can include greenhouse gas emissions, carbon footprint data, water use, and energy consumption.

The ultimate responsibility for measuring and owning these goals rests with the sustainability team, who aligns with the company's sustainability strategy set by the board.

Sustainability for your company's brand

There's a demand that customers, their executives and stakeholders, and their investors provide greater transparency into the environmental impact for the company from an operational point of view. Proactively working toward sustainability helps you establish a healthy brand for your company.

A sustainable business model can improve reputation, build trust and attract new talent/partners/investors. You can also see the benefits of sustainability as part of your business signature, combining your sustainability exercises with the capability to communicate the reasoning behind your green journey further.

Sustainability is critical to ensure long-term success as a player in the dynamic business space. Transforming the company towards a green model can also enable people's intrinsic motivation and attract new talent with an active interest in sustainability in the end.

Sustainability is an approach beyond a marketing play, with a conscious business purpose that can be part of a customer's brand identity. See [Improve customer experience and engagement](#) in the Cloud Adoption Framework.

- Add sustainability and environmental responsibility to the mission statement, ensuring that brand value alignment with the business activities and practices.
- Additionally, you should evaluate the end-to-end supply chain to ensure [Scope 3 is also measured](#).
- Ensure transparency on your commitments and progress through publicly exposing annual sustainability reports. Read more in Microsoft's [2021 Environmental Sustainability Report](#).
- Be consistent in the communication of promoting your green efforts, thus ensuring maintaining consistent brand integrity.

Regulatory compliance

Customers are responsible for ensuring they're up to date with regulatory compliance regulations across their industry. Additionally, consider how government regulations or policies for your region might influence your organization's motivations for migrating to the cloud.

A few governmental sustainability goals include:

- The UK Government targets bringing all greenhouse gas emissions to net zero by 2050.
- The US Government target to achieve net zero by 2050.

- A European Climate Law with a 55% reduction target in greenhouse emissions by 2030 and the European Union target to achieve net zero by 2050.

Understanding your current emissions

As defined by the Greenhouse Gas (GHG) Protocol, the operational boundaries of a carbon emissions inventory are broken down into three "scopes" (both direct and indirect) of emissions data, each further broken down into distinct emission sources:

- **Scope 1:** Emissions that your organization produces directly (such as using carbon-based fuels).
- **Scope 2:** Emissions that your organization incurs indirectly through purchasing electricity, heat, or steam.
- **Scope 3:** Emissions that your organization incurs indirectly beyond Scope 2 emissions (for example, emissions related to your supply chain, waste disposal, business travel, and employee commuting).

Your sustainability journey starts with working towards understanding the factors contributing to the three scopes, followed by measuring and tracking the organization's progress with each category.

- Read additional information about [Greenhouse gas emissions](#) to gain insights into the contributors to your organization's emissions.
- Read about [measuring and tracking carbon impact in the Azure Well-Architected Framework](#).

Sustainability tools and resources

To help you build your strategy Microsoft has a range of tooling and resources available.

Microsoft Cloud for Sustainability

[Microsoft Cloud for Sustainability](#) empowers organizations to accelerate sustainability progress and business growth by bringing together a set of Environmental, Social, and Governance (ESG) capabilities across the Microsoft cloud portfolio plus solutions from our global ecosystem of partners.

- Learn more by reading [what is Microsoft Cloud for Sustainability?](#)

Microsoft Sustainability Manager

The Microsoft Sustainability Manager is an extensible solution that unifies data intelligence and provides comprehensive, integrated, and automated sustainability management for organizations at any stage of their sustainability journey. It automates manual processes, enabling organizations to more efficiently record, report, and reduce their emissions.

Microsoft Sustainability Manager covers data input from sources beyond just Azure workloads. For example, you can connect to AWS, GCP, SAP, and more.

Azure carbon optimization

Use [Azure carbon optimization](#) to measure and minimize the carbon impact of your Azure resources. With Carbon optimization, you can find opportunities to optimize resource utilization to lower carbon emissions and costs, track and analyze emissions associated with Azure resources and subscriptions, and access carbon data and insights through APIs and exports.

Carbon optimization provides emission data for all Azure resource types, based on billing and usage. The emissions calculation adopts the same methodology as the Emissions Impact Dashboard and the Cloud for Sustainability API, ensuring consistency, transparency, and easy comparison of emissions data across Azure.

This capability is readily available for you in the Azure portal, and you can start using it today.

Cloud migration

Many businesses understand that [migrating workloads to the cloud](#) can cut energy consumption and costs and reduce the physical footprints of their datacenters. Transitioning workloads to Microsoft Azure can produce up to 98 percent more carbon efficiency and up to 93 percent more energy efficiency than on-premises options, depending on specific server usage, renewable energy purchases, and other factors.

Use the [Azure Migration and Modernization Program](#) to apply best practices based on the proven [Microsoft Cloud Adoption Framework](#) for Azure and [Well-Architected Framework](#) at every stage of your cloud adoption journey.

Migrate and modernize your apps, data, and infrastructure using proven cloud migration tools and patterns.

Emissions savings estimator for Microsoft Cloud

The [Emissions savings estimator for Microsoft Cloud](#) enables you to define your on-premises infrastructure workloads and then uses the information to report your current on-premises footprint and comparable Azure footprint.

Using the emissions savings estimator will help you understand the emissions-related usage of Microsoft Cloud services and estimate how much you could save by migrating to Azure.

Emissions Impact Dashboard

Once you migrate or create resources within Microsoft Azure, you'll need a reliable way of monitoring those emissions as well. The [Emissions Impact Dashboard](#) provides transparency into greenhouse gas emissions associated with using Microsoft cloud services and enables a better understanding of the root causes of emissions changes. Organizations can measure the impact of Microsoft cloud usage on their carbon footprint, and they can drill down into emissions by month, service, and datacenter region.

Microsoft's emissions reports are determined via a methodology that was validated by Stanford University in 2018, which aligns to ISO standards for measuring greenhouse gas emissions. These calculations are inclusive of Microsoft's scopes 1, 2, and 3, which means that they include emissions from Microsoft's extended network of vendors and suppliers. Reports reflect Microsoft's investments in renewables and the fuel energy mix in the regions where your computing takes place alongside other factors.

Examples of sustainability outcomes

Focusing on sustainability and protecting limited environmental resources is key to our future, and this focus also benefits business. Today, companies can draw from a broad range of assets and resources that can help them to expand into new geographic areas and develop innovative resource management solutions.

AGL, one of Australia's leading integrated energy companies, built a solution on Azure that remotely manages networked solar batteries. Learn about how the company is [growing an innovative energy partnership across Australia](#) to help local customers give back to the grid.

Bee'ah is a sustainability pioneer in the Middle East that believes in technology and sustainability creating solutions for the future. Their services include waste management, environmental consulting, renewable energy, and sustainable transportation. Azure has supported the company to launch the first AI platform to digitize all operations and

services. Read more about how the [cloud drives sustainable management and digital innovation](#) throughout the company's sustainability journey.

These customer stories demonstrate how prioritizing sustainability and environmental solutions can help organizations to create new business opportunities.

Next steps

An intentional approach can help organizations to navigate their sustainability journey. These five steps can influence outcomes for your company:

Step 1: Record and understand your company's current carbon emissions. Start by categorizing your emissions, which will help you to list of areas on which to focus.

Step 2: Evaluate whether your vendors, partners, and providers are taking steps to reduce their emissions and if these steps align with yours.

Step 3: Create an incentive for teams to reduce carbon emissions. The [Microsoft carbon fee: theory and practice](#) guide can help your organization to drive alignment and accountability across your teams.

Step 4: Seek out teams in your business to enlist their support and generate ideas for areas for improvement. Build an innovation culture where individuals are participants with a sense of ownership.

Step 5: Assess ways to incorporate [Green Software Development](#) principles into your company's sustainability goals. How can you work towards designing sustainable software applications that can be reused and has extended longevity of use and minimal computational and memory resource requirements?

Learn more about how your organization can [measure](#) and [reach](#) sustainability outcomes with the cloud.

[Reach outcomes](#)

Feedback

Was this page helpful?

 Yes

 No

How to use the business outcome template

Article • 02/28/2023

As discussed in the [business outcomes overview](#), it can be difficult to bridge the gap between business and technical conversations. This simple template is designed to help teams uniformly capture business outcomes to be used later in the development of customer transformation journey strategies.

Download the [business outcome template](#) to begin brainstorming and tracking business outcomes. Continue reading to learn how to use the template. Review the [business outcomes section](#) for ideas on potential business outcomes that could come up in executive conversations.

Use the business outcome template

In this template, business outcomes focus on three topics:

- Aligning to stakeholders or business decision makers.
- Understanding business drivers and objectives.
- Mapping outcomes to specific solutions and technical capability.

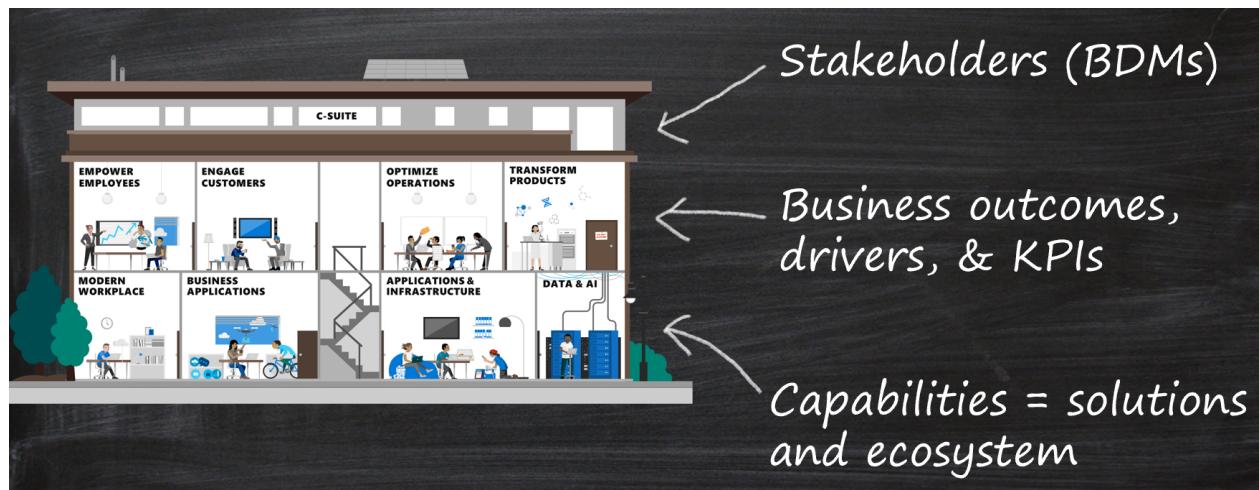


Figure 1: Business outcomes visualized as a house with stakeholders, over business outcomes, over technical capabilities.

The business outcome template focuses on simplified conversations that can quickly engage stakeholders without getting too deep into the technical solution. By rapidly understanding and aligning the key performance indicators (KPIs) and business drivers that are important to stakeholders, your team can think about high-level approaches and transformations before diving into the implementation details.

An example can be found on the "example outcome" tab of the spreadsheet, as shown below. To track multiple outcomes, add them to the "collective outcomes" tab.

XYZ Life Sciences Co. - Drug & Device Division		
Stakeholder:	Therapeutic Product Owner	Business Outcome:
Business Drivers	KPI	Capabilities
Study Design	\$3M opportunity cost per day per drug	Data-Driven protocol authoring
Study Conduct		Trial Simulation
		Structured collaboration for trial approval

Figure 2: Example of a business outcome template.

Why is this template relevant?

Discovery is a fundamental tenet of enterprise architecture. If discovery is limited to technical discovery, the solution is likely to miss many opportunities to improve the business. Enterprise architects, solution architects, and other technically minded leaders can master the discovery process by using this template. In effective discovery processes, these leaders consider five key aspects of the business outcome before leading a transformation journey, as shown in the following image:

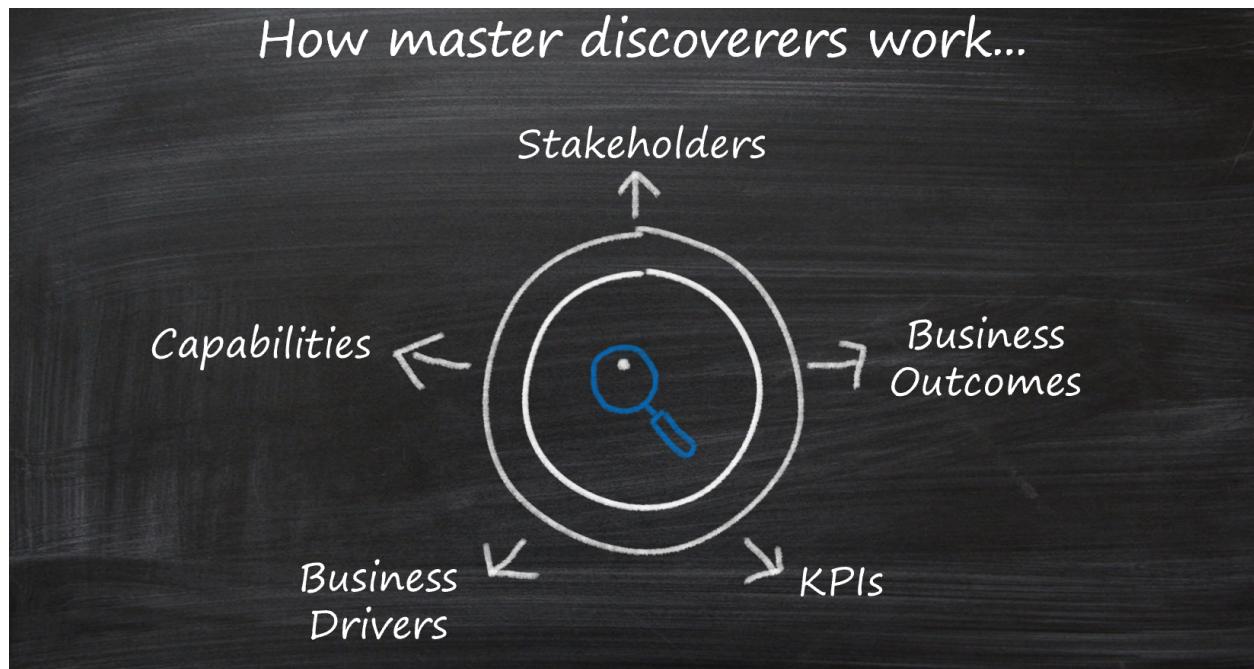


Figure 3: Five areas of focus in discovery: stakeholders, outcomes, drivers, KPIs, and capabilities.

Stakeholders: Who in the organization is likely to see the greatest value in a specific business outcome? Who is most likely to support this transformation, especially when things get tough or time consuming? Who has the greatest stake in the success of this transformation? This person is a potential stakeholder.

Business outcomes: A business outcome is a concise, defined, and observable result or change in business performance, supported by a specific measure. How does the stakeholder want to change the business? How will the business be affected? What is the value of this transformation?

Business drivers: Business drivers capture the current challenge that's preventing the company from achieving desired outcomes. They can also capture new opportunities that the business can capitalize on with the right solution. How would you describe the current challenges or future state of the business? What business functions would be changing to meet the desired outcomes?

KPIs: How will this change be measured? How does the business know whether they are successful? How frequently will this KPI be observed? Understanding each KPI helps enable incremental change and experimentation.

Capabilities: When you define any transformation journey, how will technical capabilities accelerate realization of the business outcome? What applications must be included in the transformation to achieve business objectives? How do various applications or workloads get prioritized to deliver on capabilities? How do parts of the solution need to be expanded or rearchitected to meet each of the outcomes? Can execution approaches (or timelines) be rearranged to prioritize high-impact business outcomes?

Next steps

Learn to align your technical efforts to observable and measurable outcome metrics.

[Align your technical efforts](#)

How can we align technical efforts to meaningful outcome metrics?

Article • 02/28/2023

The [business outcomes overview](#) discussed ways to communicate and measure the efficacy that digital transformation will have on your business. It could take years for your organization's business outcomes to produce measurable results. Your company's leadership could be dissatisfied with board room presentations displaying data that shows a 0 percent change for long periods.

Outcome metrics are relevant performance and impact measures collected across your organization, and address the question "*how successfully* have I achieved my change goal [x]—are these changes observable and quantifiable?" These outcome metrics can be accounted for in shorter-term increments—and then linked to progress towards your organization's longer-term business outcomes. Outcome metrics map effectively to the growth perspective of the C-suite and boardroom, and can help position company culture to become more resilient. Rather than pointing to potential lack of progress toward a long-term business goal, outcome metrics highlight early success outcomes that mark incremental progress towards long-term goals. These metrics also highlight early failure outcomes, which can produce opportunities for you to learn and adjust the strategic approach.

You're most likely already familiar with the [transformation journey](#) that most closely aligns with your desired business outcomes. To illustrate the overall concept, we provide outcome metrics for each transformation journey.

Cloud migration

This transformation focuses on cost, complexity, and efficiency, with an emphasis on IT operations. The most easily measured data behind this transformation is the movement of assets to the cloud. In this kind of transformation, the digital estate is measured by virtual machines (VMs), racks or clusters that host those VMs, datacenter operational costs, required capital expenses to maintain systems, and depreciation of those assets over time.

As VMs are moved to the cloud, dependence on on-premises legacy assets is reduced. The cost of asset maintenance is also reduced. Unfortunately, businesses can't realize the cost reduction until clusters are deprovisioned and datacenter leases expire. In many cases, the full value of the effort isn't realized until the depreciation cycles are complete.

Always align with the CFO or finance office before making financial statements. However, IT teams can generally estimate current monetary cost and future monetary cost values for each VM based on CPU, memory, and storage consumed. You can then apply that value to each migrated VM to estimate the immediate cost savings and future monetary value of the migration effort.

Application innovation

Cloud-enabled application innovation focuses largely on the customer experience and the customer's willingness to consume products and services provided by the company. It takes time for increments of change to affect consumer or customer buying behaviors. But application innovation cycles tend to be much shorter than they are in the other forms of transformation. We suggest starting with an understanding of the behaviors you wish to influence and use those behaviors as outcome metrics. In an e-commerce application, the total purchases or add-on purchases could be the target outcome, and for a video company, perhaps the time spent watching video streams.

Customer outcome metrics can be influenced by external variables, so it's important to include statistical data that's in process of being measured—release cadence, bugs resolved per release, code coverage of unit tests, page views, page throughput, page load time, and metrics relevant to application performance. Each statistic can show different activities and changes to the code base and the customer experience to correlate with higher-level patterns of customer outcomes.

Data innovation

Changing an industry, disrupting markets, or transforming products and services can take years. In a cloud-enabled data innovation effort, experimentation is key to measuring success outcomes. Be transparent by sharing prediction metrics like percent probability, the number of failed experiments, and the number of models trained. Failures will accumulate faster than successes. These outcome metrics can be discouraging, and the executive team must understand the time and investment needed to utilize them properly.

Positive outcomes associated with data-driven discovery are often: centralization of heterogeneous data sets, data ingress, and democratization of data. While your cross-functional teams continuously gather more data about tomorrow's omnichannel customer, you can produce observable results today. Supporting outcome metrics might include:

- Number of models available.

- Number of partner data sources consumed.
- Devices producing ingress data.
- Volume of ingress data.
- Types of data.

An even more valuable outcome metric is the number of dashboards created from combined data sources. This number reflects current-state business processes that are affected by new data sources. By sharing new data sources openly, your business can take advantage of the data by using reporting tools like Power BI to produce incremental insights and drive business change.

Next steps

After you align your outcome metrics, you're ready to start [measuring business outcomes](#) using objectives and key results (OKRs).

[Measure business outcomes](#)

Measure business outcomes using objectives and key results

Article • 06/23/2023

Modern operations require a modern approach to measuring business outcomes. Objectives and key results (OKR) is a powerful goal-setting framework for defining business objectives and tracking their outcomes. Your organization's OKR measurement platform should support your organization's outcomes and plan for growth by:

- Aligning the everyday work with strategic business initiatives
- Aligning the work of cross-functional teams
- Providing real-time progress and performance insights
- Enhancing staff productivity and agility

An overview of objectives and key results

The OKR framework fosters innovation, drives alignment in complex work environments, and helps individuals focus on what matters for their organization.

The two components of all OKRs are the objective and its key results. An objective is a statement of intent that describes what your team is trying to accomplish and why it's important. Key results are specific outcomes that track your team's progress towards the objective. You can define these components for your own OKRs by asking the following questions:

- **Objective:** Where do you want to go?
- **Key results:** How do you know you're getting there? How do you measure that progress?

OKRs move teams away from an "output" mindset ("What projects or tasks did we do?") to an "outcome" mindset ("What was the business result of the project or task?"). You can set OKRs at the beginning of a key time period for your business (like a quarter or a fiscal year) and then check in regularly with those OKRs to keep your team focused on the importance of their work rather than its volume.

OKR software

Organizations use specific software to leverage the OKR framework and ensure that the most important objectives are visible. Microsoft Viva Goals is a goal-alignment solution

that connects teams to your organization's strategic priorities, unites them around your mission and purpose, and drives business results.

Because Viva Goals is a part of Microsoft Viva, it integrates into the employee experience, empowering teams to be their best from anywhere.

OKRs add value to an organization

OKRs give a strategic advantage to organizations of any size and can be adopted by individuals in any role. OKRs add value to your organization in the following ways:

- **OKRs help your organization navigate rapid change, reduce risk, and identify waste.** The OKR framework gives a level of visibility into work done across your organization that allows you to proactively manage risk. It helps you identify themes, trends, and changes to diverse data sets that can go unnoticed without a unifying system. Your organization can shift focus from activity to outcomes, making the right changes at the right time to stay competitive.
- **OKRs motivate and foster involvement.** When done well and made the driving force behind your organization's strategic rhythm, OKRs can keep every employee connected to and invested in your broader mission.
- **OKRs create cross-functional cooperation that unifies initiatives and improves collaboration.** When each individual, team, and department goal is guided by the organization's broader strategy, teams can clearly prioritize each project, task, and outcome.
- **OKRs offer clear, contextual communication with continual progress reviews.** The OKR framework doesn't stop when your goals are written. Instead, it creates a strategic rhythm around the most important outcomes that your organization is pursuing. The OKR framework should influence every interaction and foster continuous improvement in every review process.
- **OKRs create clarity that is documented, measurable, and owned.** The OKR framework provides accountability that is tracked and measured. This accountability works in both directions: each individual is responsible for their OKRs, and your organization's leadership is responsible for enabling their work.

Examples of OKRs

Objectives need to inspire your organization and its teams to fully understand your mission. Key results need to be specific and measurable within each quarter. The following example OKRs can help you as you work to design your own.

Objective 1: (Product and Engineering example): Deliver a “must have” product in order to delight customers and grow our user base **Key result 1:** Increase our NPS score from 40 to 50. **Key Result 2:** Increase daily active users (DAUs) from 1,200 to 1,500. **Key Result 3:** Achieve 1,000 downloads in the app store.

Objective 2: (Human Resources example): Increase employee retention in order to do our best work. **Key Result 1:** Reduce voluntary attrition from 30% to 10%. **Key Result 2:** Increase ratio of open positions filled internally vs externally from 30% to 50%. **Key Result 3:** 100% of our employees have a standardized career plan approved by HR.

Objective 3: (Customer Success example): Optimize training process in order to increase product adoption. **Key Result 1:** Increase key account MAU from 250,0000 to 350,000. **Key Result 2:** Increase customer-facing knowledge base articles from 25 to 100. **Key Result 3:** Double participation at success office hours from 500 to 1,000 people.

Next steps

The following five steps can help your organization move forward with OKRs:

- **Step 1: Learn.** Start exploring what OKRs can do for your business. Tune in to some of your industry peers and leaders and learn how OKRs have benefited their organizations.
- **Step 2: Plan.** As you begin to draft your OKRs, ensure that your sponsors are contributing and involved in the process. Work with an OKR coach to refine your OKRs.
- **Step 3: Launch.** Each organization launches initiatives differently. Maintain a strong communication plan, and build OKR calibration and celebration process into your operating model.
- **Step 4: Drive.** Maintain rigor and focus by making sure that you share outcomes and results across your organization.
- **Step 5: Improve.** Continue to improve, revisit, and rethink how to connect more across your organization. OKRs in spreadsheets can be useful, but your organization will benefit most when everyone works together to meet objectives and gains insights from their aligned data.

[Learn more about Microsoft Viva Goals.](#)

Now that you've measured your business outcomes, you're ready to [understand financial considerations](#).

[Understand financial considerations](#)

Understand financial considerations

Article • 02/28/2023

A digital transformation goes hand-in-hand with a financial transformation. When you're making the shift to the cloud, there are financial considerations around how the cloud will impact your financial position, accounting KPIs, and processes that Chief Financial Officers (CFO) and finance teams need to understand. Your organization's motivations and business outcomes will help determine how you look at your financials.

This guidance will help you learn how to use the cloud to make your IT cost structure more flexible and help you build a business case to migrate to the cloud.

How does cloud pricing work?

The cloud uses a pay-for-what-you-consume model versus the up-front server infrastructure and software licensing costs that you would typically pay on-premises in your data center. The cloud allows you to take advantage of a variable cost model instead of a fixed cost model.

CAPEX to OPEX

One benefit of moving to the cloud is that it shifts how you pay for capacity, from CAPEX to OPEX. Your overall budget allocation moves from a CAPEX investment to OPEX pricing models that can fluctuate based on capacity or utilization of the cloud environment. Your organization will realize meaningful improvements in financial statements, with improved cash flow timing and a reduced need to acquire assets that result in a fixed cost structure.

Reduced data center footprint

A reduced data center footprint is another benefit of a shift to the cloud. On-premises data centers are frequently overbuilt for peaks, resulting in excess capacity and excess spend. In the cloud, you can spin up resources quickly with the flexibility of scalability, where you can scale up and down as needed. Azure operates in many different geographic regions, giving you flexibility to choose where to build your applications. Amortized facilities or co-lo or hosting agreements that you have in place are no longer required.

Increased productivity and service delivery

How can the cloud help your team be more productive? It's critical to consider how DevOps and the cloud can improve and drive efficiency in your organization. The cloud can help unlock better DevOps processes, resulting in faster time to market, increased revenue, increased employee productivity, and the ability to deploy products more quickly.

Sustainability

Migrating to the cloud can [reduce your carbon footprint](#), and your organization immediately benefits from a sustainability perspective. Cloud providers continuously invest in new green technologies at a dynamic scale that an organization can benefit from and decreases its environmental impact.

You can [use cost as a carbon proxy](#). The ability to track your carbon footprint across your cloud environments can sometimes be a challenge. You don't always get much visibility of the underlying infrastructure you deploy, and you can have third parties controlling some of your deployments. At some level, all your compute's embodied carbon and electricity costs are factored into the cost of all your services.

Given this, optimizing the OPEX spend can be a way of optimizing carbon efficiency.

Next steps

The financial considerations guidance above, and subsequent guidance in the Strategy methodology help you build a business case to migrate to the cloud. Before building your business case, review common finance terms that can help.

[Glossary of common finance terms](#)

Glossary of common finance terms

Article • 02/28/2023

Use these common finance terms when your team is creating a cloud migration business case. These terms can help when you share your business case with a finance team.

Terms

Amortization: An expense tied to a typically intangible asset that reflects the economic usage of that asset in a particular time period. For example if you purchase a license worth \$100, you would capitalize that on your balance sheet. If you amortized it over five years, you would annually recognize an expense of \$20 per year on your income statement.

Balance sheet: A balance sheet is a financial statement that reports a company's assets, liabilities, and shareholders' equity as of a specific date.

Capital Expense (CAPEX): The upfront investment in equipment. This equipment is capitalized as an asset and put on your balance sheet.

Cash flow statement: A cash flow statement is a financial statement that summarizes the amount of cash and cash equivalents entering and leaving a company during a given period.

Cloud economics: An understanding of the benefits and costs of the cloud, and the financial impact when you start a migration from on-premises to cloud computing.

Depreciation: An expense tied to a capitalized asset, that reflects the economic usage of that asset in a particular time period. For example if you purchase a server worth \$100, you would capitalize that on your balance sheet. If you depreciated it over five years, you would annually recognize an expense of \$20 per year on your income statement.

Double mortgage period: A period when you have two sets of costs at the same time. For example, when you have both on-premises and cloud costs.

Earnings before interest, taxes, depreciation, and amortization (EBITDA): A performance indicator of the profitability of a business. This starts from *operating income*, which is the income from your ongoing business operations (ignoring things like taxes or interest expense), and then adds back depreciation and amortization. While EBITDA is useful for comparability, it's often viewed in conjunction with metrics like capital expenditure to enable a fuller understanding of a company's ability to generate free cash flow.

Net Present Value (NPV): An assessment of the financial value of a business investment. This metric looks at cash flows, timing, and the required interest rate.

Operating Expense (OPEX): The ongoing expenses for a business. For example, a maintenance payment or periodic bill for Azure services.

Profit and Loss (P&L): A financial statement that summarizes the revenues, costs, and expenses incurred over a specified period, usually a fiscal quarter, or year. It is also referred to as the income statement.

Return on Investment (ROI): Return on investment (ROI) is a metric used to understand the profitability of an investment. ROI compares how much you paid for an investment to how much you earned to evaluate its efficiency.

Next steps

Learn more about the most common cloud accounting models for IT.

[Understand cloud accounting](#)

What is cloud accounting?

Article • 05/07/2024

The cloud changes how IT accounts for costs, as is described in [Create a financial model for cloud transformation](#). Various IT accounting models are much easier to support because of how the cloud allocates costs. So it's important to understand how to account for cloud costs before you begin a cloud transformation journey. This article outlines the most common cloud accounting models for IT.

Traditional IT accounting (cost center model)

It's often accurate to consider IT a cost center. In the traditional IT accounting model, IT consolidates purchasing power for all IT assets. As we pointed out in the [financial models](#) article, that purchasing power consolidation can include software licenses, recurring charges for CRM licensing, purchase of employee desktops, and other large costs.

When IT serves as a cost center, the perceived value of IT is largely viewed through a procurement management lens. This perception makes it difficult for the board or other executives to understand the true value that IT provides. Procurement costs tend to skew the view of IT by outweighing any other value added by the organization. This view explains why IT is often lumped into the responsibilities of either the chief financial officer or the chief operating officer. This perception of IT is limited and might be shortsighted.

Central IT accounting (profit center model)

To overcome the cost center view of IT, some CIOs opted for a centralized IT model of accounting. In this type of model, IT is treated like a competing business unit and a peer to revenue-producing business units. In some cases, this model can be entirely logical. For example, some organizations have a professional IT services division that generates a revenue stream. Frequently, centralized IT models don't generate significant revenue, making it difficult to justify the model.

Regardless of the revenue model, centralized IT accounting models are unique because of how the IT unit accounts for costs. In a traditional IT model, the IT team records costs and pays those costs from shared funds like operations and maintenance (O&M) or a dedicated profit and loss (P&L) account.

In a central IT accounting model, the IT team marks up the services provided to account for overhead, management, and other estimated expenses. It then bills the competing business units for the marked-up services. In this model, the CIO is expected to manage the P&L associated with the sale of those services. This can create inflated IT costs and contention between central IT and business units, especially when IT needs to cut costs or isn't meeting agreed-upon SLAs. During times of technology or market change, any new technology would cause a disruption to central IT's P&L, making transformation difficult.

Chargeback

One of the common first steps in changing IT's reputation as a cost center is implementing a chargeback model of accounting. This model is especially common in smaller enterprises or highly efficient IT organizations. In the chargeback model, any IT costs that are associated with a specific business unit are treated like an operating expense in that business unit's budget. This practice reduces the cumulative cost effects on IT, allowing business values to show more clearly.

In a legacy on-premises model, chargeback is difficult to realize because someone still has to carry the large capital expenses and depreciation. The ongoing conversion from capital expenditures to operating expenses associated with usage is a difficult accounting exercise. This difficulty is a major reason for the creation of the traditional IT accounting model and the central IT accounting model. The operating expenses model of cloud cost accounting is almost required if you want to efficiently deliver a chargeback model.

But you shouldn't implement this model without considering the implications. Here are a few consequences that are unique to a chargeback model:

- Chargeback results in a massive reduction of the overall IT budget. For IT organizations that are inefficient or require extensive complex technical skills in operations or maintenance, this model can expose those expenses in an unhealthy way.
- Loss of control is a common consequence. In highly political environments, chargeback can result in loss of control and staff being reallocated to the business. This could create significant inefficiencies and reduce IT's ability to consistently meet SLAs or project requirements.
- Difficulty accounting for shared services is another common consequence. If the organization has grown through acquisition and is carrying technical debt as a result, it's likely that a high percentage of shared services must be maintained to keep all systems working together effectively.

Cloud transformations include solutions to these and other consequences associated with a chargeback model. But each of those solutions includes implementation and operating expenses. The CIO and CFO should carefully weigh the pros and cons of a chargeback model before considering one.

Showback or awareness-back

For larger enterprises, a showback or awareness-back model is a safer first step in the transition from cost center to value center. This model doesn't affect financial accounting. In fact, the P&Ls of each organization don't change. The biggest shift is in mindset and awareness. In a showback or awareness-back model, IT manages the centralized, consolidated buying power as an agent for the business. In reports back to the business, IT attributes any direct costs to the relevant business unit, which reduces the perceived budget directly consumed by IT. IT also plans budgets based on the needs of the associated business units, which allows IT to more accurately account for costs associated to purely IT initiatives.

This model provides a balance between a true chargeback model and more traditional models of IT accounting.

Impact of cloud accounting models

Ensure that you choose the proper accounting model for your system design because it can affect subscription strategies, naming standards, tagging standards, and policy designs.

After you've worked with the business to make decisions about a cloud accounting model and [global markets](#), you can learn more about how to [achieve more with your investment in the cloud](#).

Next step

Achieve more with your investment in the cloud

Feedback

Was this page helpful?

Yes

No

Understand technical considerations

Article • 02/28/2023

When you're making the shift to the cloud, there are technical considerations around how it will help improve how you manage and maintain your cloud and workloads. This guidance will help you discover the technical flexibility, efficiencies, and capabilities that aren't possible with your on-premises IT infrastructure and help you build a business case to migrate to the cloud.

Technical benefits

Scalability

Scalability, or the ability to scale out your resources depending on usage, utilization, and demand, is one of the foremost technical benefits of moving to the cloud.

Right-sizing resources and utilizing auto-scaling, matching the scalability needs, turning off workloads outside of business hours and scaling only when necessary have the added benefit of driving more sustainable workloads. Read more about the [right-sizing recommendations](#) in the Azure Well-Architected Framework Sustainability workload guidance.

Availability

On-premises, it's more costly to build highly available infrastructure. It's less costly to architect highly available infrastructure in the cloud.

Security and compliance

When it comes to security and compliance, Microsoft is continually expanding our security infrastructure and toolsets to keep you on par with what's transpiring with respect to security threats on global networks.

Capacity optimization

Capacity optimization, where you only pay for the resources you utilize over time, is another technical benefit of the cloud. The core concept to consider is how elasticity and on-demand resources help you deploy, provision, or deprovision resources more dynamically.

Sustainable IT infrastructure

Building a sustainable IT infrastructure is tightly connected to moving workloads from on-premises to the cloud. So how do we select the first project to migrate based on its potential sustainability impact?

Moving the first workload is usually [aligned with the business goals](#), depending on how the migration aligns with an array of factors that lead to this decision.

Although sustainability is a key concept, in reality, it might be idealistic to believe it will be the primary decision factor for cloud migration. However, it is a significant added benefit and part of the continuous environment optimization efforts that align with the green strategy of a business.

Projected business outcomes can be synchronized with the [sustainability benefits](#) brought by the migration.

Resource-intensive workloads can be discovered using [Azure Migrate](#).

Customers can also benefit from taking the [Strategic Migration Assessment](#), analyzing the diversity of their workloads and the potential Azure destination mapping.

Depending on the results and underlying complexity, one can define migration waves.

Assess the carbon intensity using the [Emissions savings estimator](#), which supports comparing the existing footprint of a workload compared to one in Azure, based on [The Carbon Benefits of Cloud Computing: a Study of the Microsoft Cloud](#).

Identifying the first workload targeted for migration from a sustainability standpoint can be performed using the 80:20 rule – 80% of the emissions derive from 20% of the workload. This is a generic suggestion, of course, as a customer needs to perform their assessment based on the unique needs of its landscape and desired business outcomes.

Learn more about [building a sustainable IT Infrastructure](#)

Get the most out of your investment

Achieve more with your investment

With a shift to the cloud, it's important to think differently about how you'll consume and manage your cloud resources. As you build your business case, it's critical to understand the fundamental principles of cloud economics. When you plan short-term and long-term cloud solutions and align them to business outcomes, you can achieve more with every dollar you invest.

Align your partner strategy

The Cloud Adoption Framework approaches cloud adoption as a self-service activity. The objective is to empower each team, supporting adoption through standardized approaches. You can't assume that a self-service system will be sufficient for all adoption activities.

Successful cloud adoption programs typically involve at least one level of support. Some cloud adoption efforts require support from multiple partners working together towards a common goal.

Next steps

Learn how you can achieve more with your Azure investment.

Achieve more with your investment in the cloud

Achieve more with your investment in the cloud

Article • 02/28/2023

When migrating to the cloud, it's important to think differently about how you'll consume and manage your cloud resources. As you build your business case, it's critical to understand the key principles of cloud economics and transform your mindset. Part of this transformation is discovering technical and financial flexibility, efficiencies, and capabilities that aren't possible with your on-premises IT infrastructure. When you plan short-term and long-term cloud solutions and align them to [business outcomes](#), you can achieve more with every dollar you invest.

How does cloud pricing work?

Cloud costs are tied to compute and storage and include the underlying software licensing fees. The cloud uses a pay-for-what-you-consume model versus the up-front server infrastructure and software licensing costs that you would typically pay on-premises in your data center. On-premises, you typically have a combination of upfront costs and operating expenditures. When you move to the cloud, you shift to the pay-as-you-consume model and mainly operating expenditures.

To take advantage of the best pricing in the cloud, you must understand how you'll consume resources for your specific workloads. Once you have a consumption plan, you can establish your fixed and variable cost models to maximize your investment.

Understand your workloads

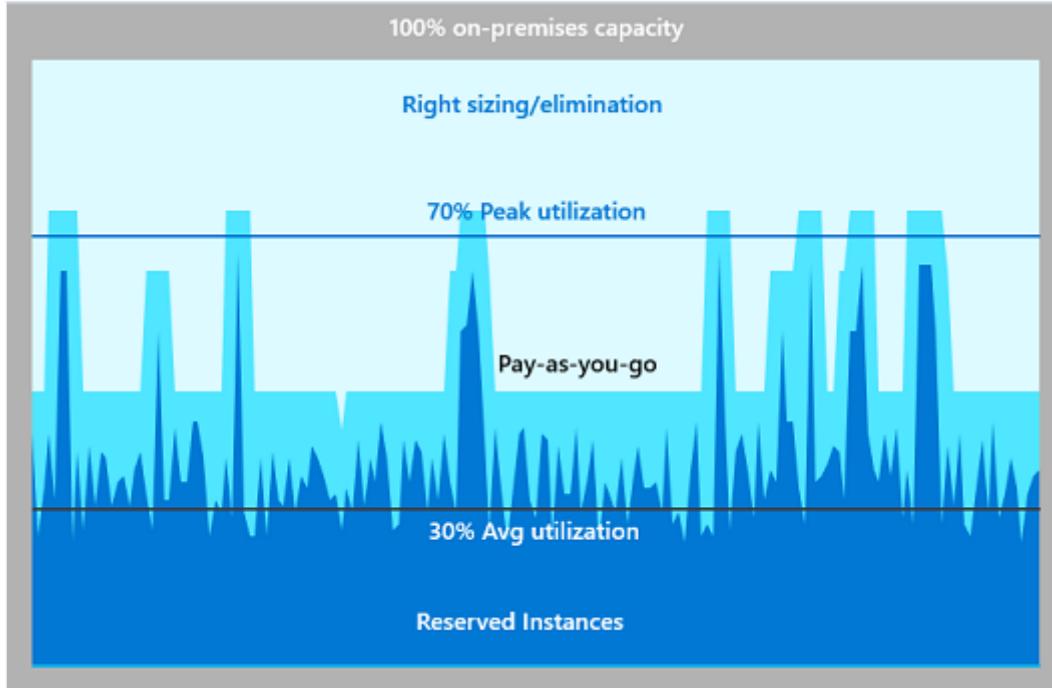
On-premises, your architecture is typically provisioned for peak capacity. Migrating from on-premises to the cloud gives you the flexibility of scalability, and you scale up and down as needed. It's critical to understand your workloads to realize the full benefits of the cloud.

Idle capacity: Azure helps eliminate the idle capacity created by overprovisioned architecture for coverage during peak usage. Rightsizing and eliminating workloads you don't need helps reduce your idle capacity when moving to the cloud. This exercise delivers immediate savings and cash flow reductions.

Unpredictable workloads: You can scale your compute resources up and down in the cloud as the demands of your business change. You're able to scale your capacity up

and down and use a variable cost model as opposed to a fixed cost model. This elasticity of the cloud makes the pay-for-what-you-consume model possible and works well for your unpredictable workloads. Consider using virtual machine scale sets and snoozing VMs to only pay for the resources you need when you need them.

Predictable workloads: For your predictable workloads, you can take advantage of the cost-savings offers such as Azure Reservations.



Initial clean-up, right-sizing, and optimization: When planning to move to Azure, review which workloads are no longer needed. This process of clean-up can help you build a stronger business case and show an immediate effect on your budgets. For workloads you still want to use and bring to the cloud, you can use tools to help optimize them, for example, Azure Migrate.

Take advantage of cost-savings offers

There are many cost-savings offers that can help reduce your cloud costs:

Azure Hybrid Benefit: Reduce the costs of running your workloads in the cloud by using this licensing benefit. You can use your on-premises Software Assurance-enabled Windows Server and SQL Server licenses on Azure. This benefit applies to RedHat and SUSE Linux subscriptions, too. To learn more, see [Azure Hybrid Benefit](#).

Spot virtual machines: You can use spot virtual machines with deep discounts for workloads that can be interrupted and don't need to complete within a specific time frame. For example, high-performance computing scenarios, batch processing jobs, visual rendering applications, dev/test environments, including continuous integration

and continuous delivery workloads, or large-scale stateless applications. To learn more, see [Spot virtual machines](#).

Reservations: Receive a discount on your workloads when you reserve your resources in advance. In return, Microsoft passes the savings onto you as discounts of up to 72 percent.¹ For more information, see [Azure reservations](#).

Azure Dev/Test pricing: Take advantage of discounted rates for your development and testing, including the Microsoft software charges on Azure Virtual Machines and special dev/test pricing on other services. For more information, see [Azure Dev/Test pricing](#).

Extended security updates: Receive continued support for SQL Server 2008 and SQL Server 2008 R2 in the cloud, which has reached the end of their support lifecycle. You can migrate your on-premises SQL Server instances to Azure Virtual Machines, Azure SQL Database, or stay on-premises and purchase extended security updates. You'll receive free extended security patches by migrating to an Azure Virtual Machine. To learn more, see [Lifecycle FAQ: Extended Security Updates](#).

Continually optimize your environment

Microsoft provides frameworks and tools to help you understand your costs and continually optimize your environment:

Understand and forecast your costs: Monitor your bill, set budgets, and allocate costs to teams and projects with Azure Cost Management + Billing.

Learn more:

- [Optimize costs from recommendations](#).
- [Prevent unexpected charges](#).

Cost optimize your workloads: Optimize your resources and architecture with Azure best practices from Azure Advisor and the Microsoft Azure Well-Architected Framework.

Learn more:

- Read about [Azure Advisor](#).
- Get [Azure Advisor cost recommendations](#).
- Learn about the [Microsoft Azure Well-Architected Review](#).
- Learn about the [Microsoft Azure Well-Architected Framework](#).

Save with Azure offers and licensing terms like the Azure Hybrid Benefit and Azure Reservations.:.

Learn more:

- Learn about the [Azure Hybrid Benefit](#).
- Learn about [Azure Hybrid Benefit for Windows Server](#).
- Review [pricing guidance for SQL Server Azure VMs](#).
- Learn about [Azure Reservations](#).
- Read the [reserved instances FAQ](#).

Control your costs: Establish spending goals and policies with guidance from the Microsoft Cloud Adoption Framework for Azure. Implement cost controls with Azure Policy so your teams can go fast while complying with policy. For more information, see [Enforce tagging conventions using Azure Policy](#).

Understand your financial stories

The core financial benefits of Azure are driven by a fundamental shift in the IT operating model. This shift benefits your organization's core financial statements and frees up cash flow for reinvestment:

Balance sheet: When you operate on-premises in datacenters, you typically have invested up front in long-term assets that limit the cash and capital required to grow your business. While in the cloud, you can shift datacenter operations costs into modernization, developing cloud applications, and other projects that drive business growth. This shift can make your balance sheet more agile.

Cash flow statement: The pay-for-what-you-consume model and the ability to apply policies and tags to your Azure resources help you improve the predictability of your cash flow statement. This model delays spend and improves your cash flow timing.

Income statement (profit and loss): You can improve profitability over time by taking advantage of Azure's flexibility, low management costs, services, and pricing models.

Achieve more with your investment

The goal of your cloud business case is to achieve more with every dollar invested. This goal can be accomplished by releasing committed cash flows and budgets that can be reinvested into further modernization. This concept is the velocity of the dollar - you accelerate value per dollar through phased reinvestment driving modernization and value.

The initial technical benefits of a cloud migration focus on the lift and shift model, where you migrate workloads to infrastructure-as-a-service (IaaS) in the cloud. The goal is to

get the most out of your on-premises investment, and then move those workloads to IaaS, potentially freeing up cash flow. Historically, this process would be considered a savings opportunity. This approach in the cloud is better viewed as a *reinvestment opportunity*.

As you free up cash flow, continue your cloud adoption, and mature your workloads, you can reinvest the savings to modernize to different service levels. Once you have your initial workloads in IaaS, you might consider shifting some workloads to platform-as-a-service (PaaS). You'll still provide the same type of service delivery however, you'll deliver it at a lower cost with more features and functionality. The next step in the iterative modernization process is moving some workflows and line-of-business applications to software-as-a-service (SaaS).

When you plan your phases of cloud maturity and the reinvestment of your cloud savings, and align with [business outcomes](#), you can achieve more with every dollar you invest.

Next steps

Learn more about how to align your partner strategy when you migrate to the cloud.

[Strategy for partner alignment](#)

Strategy for partner alignment

Article • 07/18/2023

The Cloud Adoption Framework approaches cloud adoption as a self-service activity. The objective is to empower each of the teams supporting adoption through standardized approaches. In practice, you can't assume that a self-service approach will be sufficient for all adoption activities.

Successful cloud adoption programs typically involve at least one level of support. Some cloud adoption efforts may require support from multiple partners working together towards a common goal.

Steps to align the partnership strategy

It's important to start aligning your partnership strategy during the strategy definition phase of adoption. The following steps help remove roadblocks in later phases of the adoption lifecycle.

1. Start to understand support needs.
2. Consider partnership options that fit your culture and needs.
3. Evaluate a shortlist of partner options.
4. Begin contract and paperwork reviews with selected partners.

Completing these steps early, will ensure success of the team when the technical efforts begin. The following sections of this article provide guidance for each of these steps.

Understanding support needs

Throughout the cloud adoption lifecycle, the various teams may require support to be successful. The following are a few examples of the types of help commonly required.

- **Strategy:** Support defining the business strategy, building a business case, and supporting technology strategy.
- **Plan:** Support with discovery of the portfolio, quantitative assessment of the digital estate, development of a cloud adoption plan, creation of a skilling plan.
- **Ready:** Support deploying a landing zone or full cloud environment capable of supporting the cloud adoption plan.
- **Migrate:** Assistance migrating workloads or building a migration factory to ensure sound migration processes.

- **Innovate:** Assistance developing new solutions or rebuilding/rearchitecting existing solutions to drive innovation.
- **Govern:** Support or ongoing managed services to provide governance and controls across the cloud environment.
- **Manage:** Support or ongoing managed services to operate the cloud platform and the workloads hosted in the cloud.

Few corporations have the diversity of skills required to support strategy, planning, readiness, adoption, governance, and management. Partners and other support models are often necessary to fill in the gaps in the team's skills and responsibilities.

Various partnership options can help develop needed skills, augment staffing requirement, or completely offload-specific processes.

Partnership options

You are not alone in your cloud journey. There are several options to support your team throughout your cloud adoption journey.

- **Azure solution providers (partners):** Get connected with Azure expert-managed services providers (MSP) and other Microsoft partners who have service offerings aligned to the Cloud Adoption Framework methodologies.
- **FastTrack for Azure:** Use the Microsoft FastTrack for Azure program to accelerate migration.
- **Azure Migrate and Modernize:** Azure Migrate and Modernize aligns a mixture of partners and Microsoft employees to accelerate and support your migration.
- **Solution assessments:** Get assistance from a Microsoft solutions assessment expert or qualified partner as part of a [Solution Assessment Program engagement](#).

Azure solution providers

Microsoft certified solution providers specialize in providing modern customer solutions base on Microsoft technologies across the world. Optimize your business in the cloud with help from an experienced partner.

[Find a Cloud Solution Provider \(CSP\)](#). A certified CSP can help take full advantage of the cloud by assessing business goals for cloud adoption, identifying the right cloud solution that meets business needs and helps the business become more agile and efficient.

Azure expert-managed services providers (MSP) have undergone a third-party audit to validate a higher tier of capability, demonstrated through certified staff headcounts,

customer references, annual consumption of Azure at scale, and other criteria.

[Find a managed service partner](#). An Azure managed service partner (MSP) helps a business transition to Azure by guiding all aspects of the cloud journey. From consulting to migrations and operations management, cloud MSPs show customers all the benefits that come with cloud adoption. They also act as a one-stop shop for common support, provisioning, and the billing experience, all with a flexible pay-as-you-go business model.

In parallel to the development of the cloud adoption strategy, the cloud strategy team should start to identify solution providers that can partner in the delivery of business objectives.

FastTrack for Azure

[FastTrack for Azure](#) provides direct assistance from Azure engineers, working hand in hand with partners, to help customers build Azure solutions quickly and confidently. FastTrack brings best practices and tools from real customer experiences to guide customers from setup, configuration, and development to production of Azure solutions, including:

During a typical FastTrack for Azure engagement, Microsoft helps to define the business vision to plan and develop Azure solutions successfully. The team assesses architectural needs and provides guidance, design principles, tools, and resources to help build, deploy, and manage Azure solutions. The team matches skilled partners for deployment services on request and periodically checks in to ensure that deployment is on track and to help remove blockers.

Azure Migrate and Modernize

[Azure Migrate and Modernize](#) provides a mixture of technical skill building, step-by-step guidance, free migration tools, and potential offers to reduce migration costs.

Microsoft uses FastTrack for Azure and Azure solution providers to improve customer success during migration.

Watch this short video of how Azure Migrate and Modernize can help you:

<https://www.microsoft.com/en-us/videoplayer/embed/RE4D1vH?postJs||Msg=true>

Solution assessments

Get assistance from a Microsoft solutions assessment expert or qualified partner as part of a [Solution Assessment engagement](#). Microsoft Solution Assessments provide customers with an in depth understanding of the opportunities available in their environments to improve productivity, reduce cost and optimize investments. These assessments utilize modern tools to collect the customer's data estate, analyze the deployed environment and provide insights for data-driven recommendations to help customers determine actionable steps for digital transformations, cloud migrations and process improvement.

Azure support

If you have questions or need help, [create a support request](#). If your support request requires deep technical guidance, visit [Azure support plans](#) to align the best plan for your needs.

Shortlist of partner options

During strategy development, it's hard to define specific partnership needs. During development of the cloud adoption plan and skilling plan, those needs will come into focus.

But, based on the culture and maturity of your team, it may be possible to decide on a partnership option that is more aligned with your expected needs.

Choose one or more of the partnership options above to narrow down the options to investigate first.

Begin contract and paperwork reviews

As the shortlist of options is reviewed, there will likely be one or more partners that stand out. If there is a clear leader among the partners, start the process to review contracts and paperwork with the partner.

The contracting process can take time. Reviewing legal terms ahead of time can remove one barrier to engagement when your teams need help the most.

This is especially true if your company requires vendors to be added to an approved vendor list.

Next steps

After your partner alignment strategy is kicked off, you may want to consider your [security strategy](#) next.

[Define your security strategy](#)

Build a digital transformation timeline

Article • 02/28/2023

Digital transformation and cloud adoption might seem like technical project. But at their core, both are complex change management efforts involving multiple stakeholders moving in alignment to accomplish business and technical change. When they move in unison, the impact on the business is apparent.

Competing objectives

If you've been building out the [strategy and plan template](#) for your organization, some trends have likely begun to emerge:

- Competing motivations: Your organization is likely motivated by needs that span several of the [motivation categories](#): Migration motivations, innovation motivations, and critical business events. Each of these categories requires slightly different behaviors during technical implementation and business adoption.
- Competing business outcomes: Your organization is also likely to see competing [business outcomes](#). It's possible to improve customer engagements in the cloud. You could also use the cloud to achieve cost reduction outcomes. But the behaviors required to do each are different, making it difficult to succeed at both in parallel.
- Aligning financial and technical considerations: If motivations and outcomes aren't aligned, it's easy to have misaligned conceptions when evaluating key considerations. This can lead to difficulty executing your strategy or at minimum communicating success.

It's possible to achieve conflicting motivations and competing outcomes. However, your successes will be greater if you prioritize your transformation and develop a timeline that focuses on the most important objectives first.

Constrained staff

Change management of any type places a strain on existing staff. Cloud adoption is no different. Even if the financial and technical considerations look good on paper, competing or parallel outcomes will intensify the strains of change management.

Even with adequate staffing, or ample budget for implementation partners, staffing constraints are a notable risk when outcomes are expected in parallel. There's typically some overlooked shared dependency, which becomes a blocker for one or more

projects. Usually, those overlooked dependencies follow Murphy's law and become an issue when they are on the critical path for multiple projects.

If you must deliver parallel outcomes, scrutinize your staffing plans and critical path for any potential shared dependencies before you commit to the strategy.

De-risk with cloud adoption horizons

It's common for organizations to have competing motivations. But competing motivations create a divide in the financial investments that the company can make in any one objective. Such a divide leads to an overall reduction in the amount of change or transformation that any effort can deliver.

Stratification is a sound principle in macroeconomics. But with the limited budget of most technology-driven change management projects, a stratified approach leads to confusing and distracting signals within the program. More noticeably, multiple, simultaneous investments in competing strategies lead to misalignment of the people, processes, and projects that are required for overall program success. For digital transformation to succeed, organizations must prioritize motivations based on timeline expectations, organizational alignment, and capacity for investment.

To create clarity and alignment, it's suggested that complex digital transformation projects align to an organization's horizons or phased program delivery. In this type of approach, the company commits to a single motivation category for a time-bound period. All teams and organizations prioritize investments and collaborate to support the priority outcome, as needed, for the defined period of time. This approach creates unity, clarity, and drives a snowball effect, allowing the success of one horizon to accelerate the target outcomes of the next horizon.

In this model, the full program team can overcome skills gaps, together. Support one another in achieving small but critical milestones, together. Deliver on those outcomes more quickly, then move on to the next horizon together. In the end, a supportive team-based approach with loosely defined and agile horizons will progress faster than the same teams competing with one another for limited resources.

Example of a cloud adoption plan with four horizons

In this example, a customer has a high number of motivations that appear in each category of the motivations table. Diverse motivations suggest that the organization has

multiple critical business events to address, a need for operational improvements from a migration or modernization, and innovation opportunities:

- **Datacenter exit:** They're delivering on a datacenter exit requiring significant focus from the central IT and cloud adoption teams. The 12- to 18-month time frame to fully plan two datacenter replacements is aggressive but realistic, if there are few distractions.
- **Operational improvements:** Accelerating innovation requires modernizing existing operational systems. It also requires modernizing processes that are dedicated to current production environments.
- **Innovation expansion:** The long-term objective is to continue to grow and lead the market through innovation. The ultimate goal is to maximize the amount of effort IT invests in innovation overall and reduce existing operational investments.

This reference customer developed horizons aligned to the following target schedule, to be evaluated and updated as part of quarterly release plan:

Horizon	Objective	Time frame	Considerations
1. Migration and modernization	Prioritize the datacenter exit with a focus on modern platform as a service (PaaS) solutions over a basic lift-and-shift migration.	Months 0 - 18	The migration as priority should minimize conflicts with existing innovation commitments.
2. Operation modernization	Prioritize operational improvements built on cloud-native governance, operations management, security, and compliance capabilities.	Months 6 - 18	This effort complements and supports the primary migration effort.
3. Advanced modernization	With post-migration and operational improvements, the team will have sufficient data and cloud skills to perform deeper modernization of complex architectures.	Months 18-24	
4. Innovation and growth	Redirect capital reduction from datacenter exits and new skills in central IT to focus on accelerating continued innovation.	Month 24+	All prior horizons will produce a long list of new innovations as the central IT and cloud adoption teams create tighter collaborations and build out automation assets.

Next steps

Developing a similar set of cloud horizons can help refine your technical and financial considerations. This approach could also make it easier to share your priorities and objectives with stakeholders and technical staff to ensure their efforts align with the desired outcomes. With defined cloud adoption horizons and updated considerations, you're now ready to [create a business case](#).

[Create a business case](#)

Create a business case for cloud migration

Article • 03/27/2023

Your organization depends on information technology (IT) for its operations, and probably for creating and supplying its products as well. It's a significant expense. Migrating IT resources to the cloud offers the potential for cost savings.

However, a move to the cloud must be carefully considered and planned. Creating a business case for cloud migration can help foster support from your Finance team and other areas of the business. It can also help accelerate cloud migration and promote business [agility](#).

A business case provides a view of the technical and financial timeline of your environment. Developing a business case includes building a financial plan that takes technical considerations into account and aligns with [business outcomes](#).

Key components of a business case

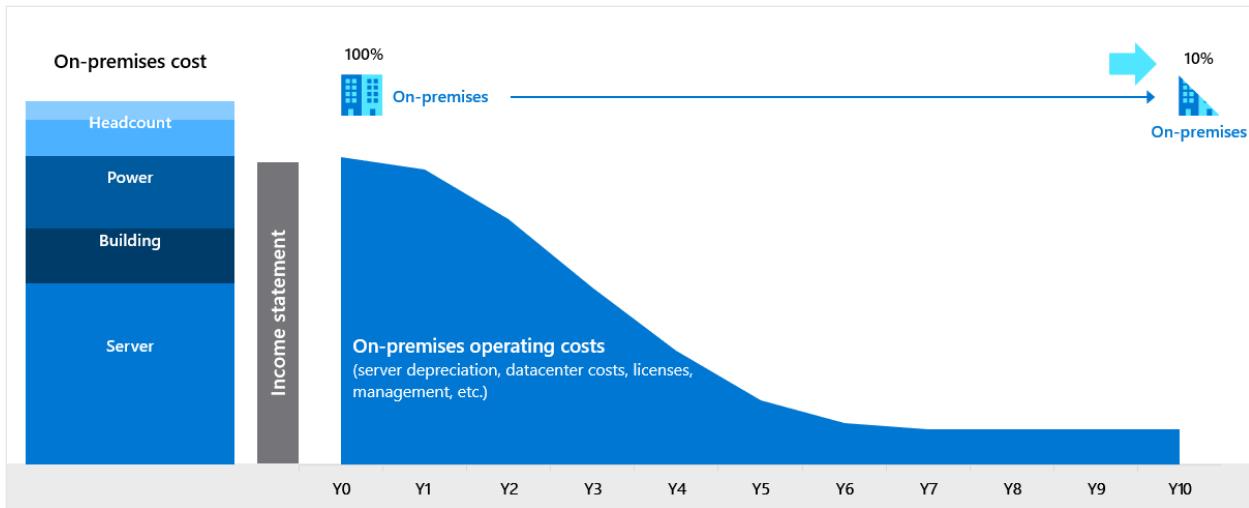
There are several key components to consider when you plan a business case.

Environment scope: As you build out the on-premises view of your environment, think about how your environment scope—from both a technical and financial perspective—is aligned. You want to be sure the technical environment you're using for your plan matches up to the financial data.

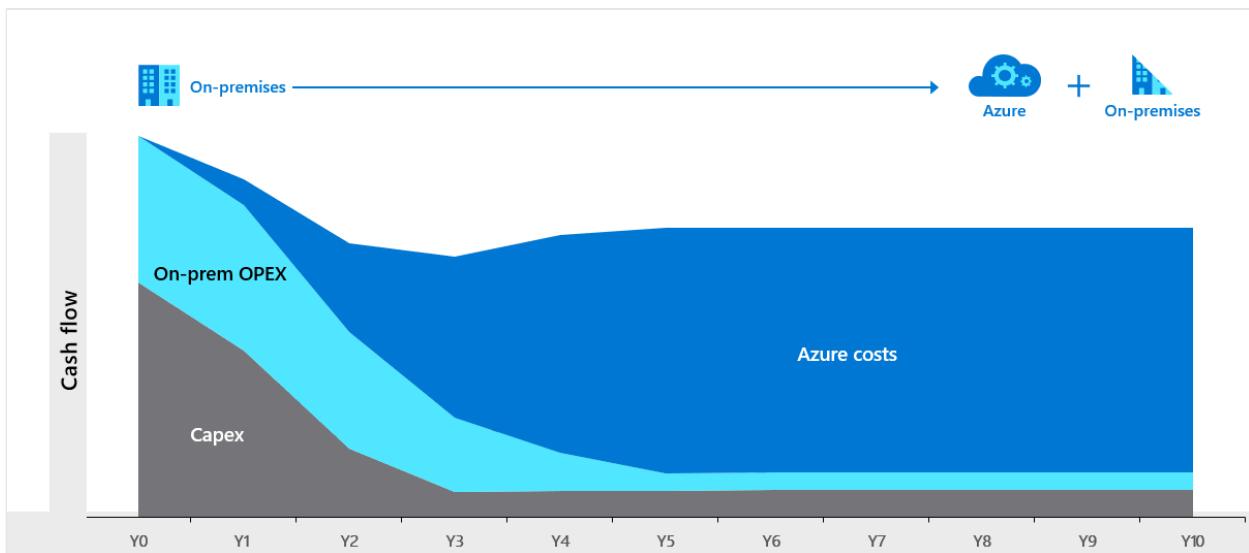
Baseline financial data: Common questions you can ask to gather needed financial data are:

- How much does it cost to run my environment today?
- What am I spending on servers in an average year?
- What am I spending in my data center operations categories, for example, power or lease costs?
- When is the next hardware refresh?

On-premises cost scenario: Forecast your on-premises costs if you don't migrate to the cloud.



Azure scenario for on-premises costs: Forecast your on-premises costs when you migrate to the cloud in an Azure scenario. It takes resources and time to shift your environment to the cloud, so it's important to account for them in the business case. Include all of the core benefits that the cloud provides.



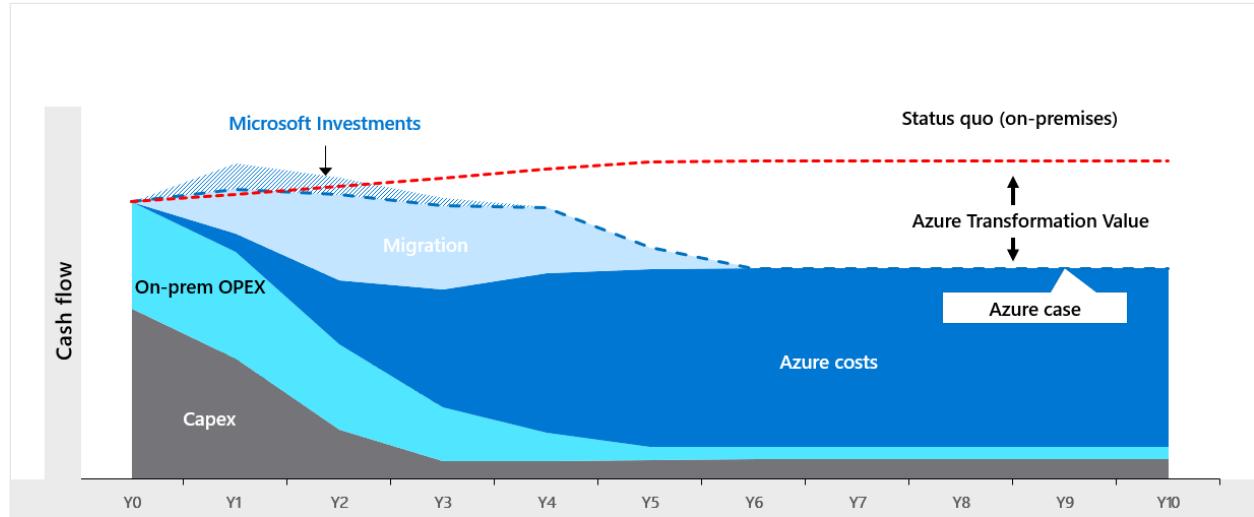
Migration timeline and Azure costs: Forecast the migration timeline and estimated costs for a given environment. Consider how you can optimize and get the most out of your Azure investment. For example, use reserved instances, scale capacity up and down, use the Azure hybrid benefit, and right-size your resources.

A business case isn't just a moment-in-time view. It's a plan that covers a period of time. As you shift to the cloud, your costs begin to decrease. You can forecast the ramp-down in on-premises spending over time associated with your cloud migration plan.

Once on-premises workloads and cost structure have been identified, you can then build out your optimized Azure consumption plan.

As a final step, compare the cloud environment to an on-premises or status quo scenario so you can assess the benefit of migrating to the cloud. The Azure view will

show reduced on-premises costs over time, your Azure environment costs, and any costs associated with the cloud migration.



Cloud savings

Migrating resources to the cloud can save organizations money. Cloud billing models differ from on-premises, creating meaningful savings opportunities to reduce costs. Savings can then be reinvested into new technology initiatives.

Cloud costs are flexible, and they can be reduced with:

Azure hybrid benefit: Reduce the costs of running workloads in the cloud by using [this licensing benefit](#). You can use your on-premises Software Assurance-enabled Windows Server and SQL Server licenses on Azure. This benefit also applies to RedHat and SUSE Linux subscriptions.

Spot virtual machines: You can use [spot virtual machines](#) with deep discounts for workloads that can be interrupted and don't need to complete within a specific time frame. For example, high-performance computing scenarios, batch processing jobs, visual rendering applications, dev/test environments including continuous integration and continuous delivery workloads, or large-scale stateless applications.

Reservations: Receive a discount on your workloads when you [reserve resources in advance](#). In return, Microsoft passes the savings onto you as discounts of up to 72 percent.

Azure savings plan for compute: Azure savings plan for compute is a flexible cost-saving plan that generates significant savings on pay-as-you-go prices with a one-year or three-year contract. Eligible compute services include virtual machines, dedicated hosts, container instances, Azure premium functions, and Azure app services. Savings apply to these compute services regardless of the region, instance size, or operating

system. To further optimize cost and flexibility, you can combine an Azure savings plan with Azure Reservations. For more information, see [Azure savings plan overview](#) and [Azure savings plan documentation](#).

Azure dev/test pricing: Take advantage of [discounted rates for development and testing](#). It includes the Microsoft software charges on Azure Virtual Machines and special dev/test pricing on other services.

Extended security updates: Receive [continued support](#) for SQL Server 2008 and SQL Server 2008 R2 even though they've reached the end of their support lifecycle. You can migrate your on-premises SQL Server instances to Azure Virtual Machines, Azure SQL Database, or stay on-premises and purchase extended security updates. You receive free extended security patches by migrating to an Azure VM.

Tools and calculators

There are many valuable tools and calculators you can use to help prepare a business case for your cloud migration.

Azure Total Cost of Ownership (TCO) Calculator: Use the [TCO calculator](#) to estimate the cost savings you can realize when you migrate your workloads to Azure.

You can enter details of your on-premises infrastructure, including servers, databases, storage, and networking, licensing assumptions and costs. The calculator creates a match from Azure Services to create a high-level initial TCO comparison. However, the results of the TCO calculator need to be considered with care, since an on-premises server list is often complex and optimization steps can be taken when considering Azure.

Retail Rates Prices API: Use the [Retail Rates Prices API](#) to retrieve retail prices for all Azure services. Previously, the only way that you could retrieve prices for Azure services was to either use the Azure Pricing Calculator or use the Azure portal. This API gives you an unauthenticated experience to get retail rates for all Azure services. Use the API to explore prices for Azure services against different regions and different SKUs. The programmatic API can also help you create your own tools for internal analysis and price comparison across SKUs and regions.

Azure VM cost estimator: This Power BI model allows you to estimate your cost savings against pay-as-you-go pricing by optimizing Azure offers and benefits for VMs like Azure Hybrid Benefit and reserved instances.

Download the following files to use the Power BI model:

- [Power BI template ↗](#)
- [Excel file ↗](#)

Azure Pricing Calculator: Use the [pricing calculator ↗](#) to estimate your hourly or monthly costs for Azure products.

Partner toolsets: Microsoft Partners have tools in the [Azure Marketplace ↗](#) that can help create a migration cost analysis.

Solution assessments: Get assistance from a Microsoft solutions assessment expert or qualified partner as part of a [solution assessment engagement ↗](#).

Azure Migration and Modernization Program: Join [this program ↗](#) to get guidance and expert help at every stage of the cloud migration journey. Migrate infrastructure, databases, and apps so you can move forward with confidence.

Learn about Azure

[Microsoft Learn training](#) offers many learning paths for Azure that you might want to consider as you build your business case.

- [Control Azure spending and manage bills with Azure Cost Management + Billing.](#)
- [Microsoft Azure Well-Architected Framework - Cost Optimization](#)
- [Plan and manage your Azure costs](#)
- [Analyze costs and create budgets with Azure Cost Management](#)
- [Configure and manage costs as a Microsoft partner by using Azure Cost Management](#)

Next steps

Learn more about how to share your strategy and business case to migrate to the cloud:

[Share your strategy](#)

Building your DevOps practice

Article • 02/28/2023

This article guides you through various considerations as you plan to build out your DevOps practice. The following sections can help you answer critical questions such as:

- Should our organization invest in building a DevOps team?
- What makes infrastructure-as-code preferable?
- How can automated workload and application deployment benefit our organization?
- In what ways does our operating model need to evolve over time?

Cloud architectures offer a new way to do business, offering benefits that include:

- Giving IT teams the ability to scale and democratize services
- Enabling global expansion opportunities
- Providing increased consistency and velocity

As your customers evolve and look to deliver differentiated services to accelerate business, use a well-designed DevOps practice in your organization to ensure your teams can collaborate effectively and build software at a higher velocity while also maintaining necessary controls, security and governance.

To have a strong cloud business, you need a mature DevOps practice. Unifying people, process, and technology helps you transition your organization's cloud practice into an effective operating model.

The cloud is an operating model, not a destination

Treating the cloud as a destination is typically an indicator that an organization hasn't yet transformed its practices and delivery of services. Organizations in this mode are often highly dependent on central IT teams for provisioning. The central IT teams then hand off the system to application teams or developers. This type of workflow can slow progress and drive up costs.

When they adopt a cloud strategy, organizations:

- Democratize IT resources
- Use more cloud native services
- Enable self-service

- Use repeatable builds
- Rely on skills both in IT teams and across the entire organization

It's not wrong to use VMs and infrastructure as a service. However, you can reduce costs and overhead and improve your service delivery if you transition to using microservices and platform services and shift the role of your central IT team to governance and empowerment.

Evolve traditional functions to make them cloud aligned

The Organize methodology of the Cloud Adoption Framework describes the functions necessary for your cloud practice. It also offers guidance on data silo avoidance and how you can empower users and tap into resources across your entire organization. Even small IT teams can accomplish a lot by removing barriers and enabling self-service for others within your organization. As you build your cloud capabilities, you'll see operations teams embracing infrastructure-as-code and traditional network, security, and system admins writing scripts to drive greater consistency.

As an example, imagine that the fictional organization Contoso Corporation acquires a new company. They need to incorporate this company, and all its applications and data, into Contoso's Azure environment. Contoso has fully scripted a new landing zone deployment. In it, they provision a subscription with appropriate permissions, network peering that connects it back to their hub network, and policies that comply with Contoso's corporate governance. The entire process takes no longer than 10 minutes before the new company can begin to migrate their applications into the new landing zone Contoso created.

In more traditional environments, the process involves a much longer waterfall approach:

- The organization acquires the hardware or service
- IT team deploys the infrastructure
- Networking configures the routing
- Security validates and applies security policies
- Governance checks compliance and applies missing policies

Each of these phases in a traditional environment can take weeks or months to complete, in contrast to the mere minutes required in a modern cloud approach.

Repeatability drives velocity

Your move to a code-based infrastructure doesn't need to be complicated. Some simple changes can carry you a long way.

Create repeatable templates to automate your workload deployments. Build golden image VMs that have your desired security parameters, storage accounts for your long term and short term storage needs, and templates for Linux-based web applications that enable SSL only.

You can house the code for these templates in various storage options, including a git repository (like Azure DevOps or GitHub) or something like version-controlled Template Specs in Azure. The particular system your organization chooses doesn't matter so much as the adoption of the process as a whole: build, code, deploy, reuse, learn.

Should I build all workloads through code?

Even if DevOps is critical for your strategy, not every workload is a good candidate. It's important that you identify early on which applications might benefit from an investment in DevOps and which applications won't offer much return on that investment. Consider the best way to manage these workloads and who you should make responsible for their management. Some examples of these applications include commercial off-the-shelf (COTS) applications, applications slated for sunsetting, and some legacy applications.

Customers are key

Listening and learning from your customers is a key to your DevOps practice. As you build and maintain your organization's DevOps practice, get feedback from your users, incorporate it into your backlog, and use it as a source for continuous improvement.

As an example, one area in our own DevOps process where customers can provide direct feedback is in our technical documentation. Each published article has an edit button at the top. Readers can edit the article, initiating a pull request that goes to a specific document's owner. The owner can then review and incorporate suggestions directly from a reader. Through this process, Microsoft is able to apply skills and expertise from our entire user base in addition to the expertise within our organization.

DevOps is more than just developers building a specific product. It's an approach involving multiple key elements of the product lifecycle—a practice known as *everything as code*.

Takeaway

As you read through the Cloud Adoption Framework documentation, notice how DevOps is present in all phases. Consider using the following process as you begin your own implementation.

1. Design your strategy with DevOps in mind.
2. Reimagine your organization as a cloud based organization.
3. Operationalize your plan using repeatable agile principles.
4. Enable CI/CD in your landing zone for consistent code based management
5. Evaluate DevSecOps to understand how security fits within your DevOps cycles.
6. Deploy your workloads through pipelines and/or actions.
7. Establish a management baseline to provide each DevOps team with consistent tools and foundations for operations.
8. Integrate insights and management into your DevOps tooling so that you can appropriately respond to changing conditions and demands.
9. Create governance guardrails so your DevOps team(s) can be confident in their ability to deploy to production without violating corporate policies.
10. Finally, consider gathering user feedback, incorporating it into your backlog, and iterating rapidly to keep your users engaged.

Next steps

[Business outcomes overview](#)

Share your strategy

Article • 02/28/2023

After your team has decided on the strategy and roadmap for your cloud adoption plan, you'll want to share your strategy with your organization's finance team and gain their support. Your goal is to showcase the benefits and cost-saving measures of moving services to the cloud from an on-premises plan.

Get started

The information in the following guidance is a reminder of tasks to complete before sharing your strategy for cloud migration with the finance team.

Gather data

Augment your business case with as much data as possible. Personalizing it specifically to your organization gives your finance team a view of how the cloud can benefit your organization. It helps to review your most recent balance sheet, income statement, and cashflow statement.

Identify a focus

What's most important to your organization? What business outcomes do you want to achieve? Cost savings and business agility are some of the topics you might focus on. Customize and align your strategy to your organization's goals, business outcomes, and greatest needs.

Use finance terminology

Use terms and acronyms familiar to the finance team, for example, capital expenditure (CAPEX), operational expenditure (OPEX), and double mortgage period. Using these terms keeps your business case streamlined and prevents over-explanation. See the [finance glossary](#) for detailed descriptions of the most common finance terms.

Share your business case

Show the finance team your business case that includes a financial roadmap and multi-year plan. Your plan should reiterate your organization's business and financial goals, and demonstrate how cloud economics aligns to their financial considerations.

Your business case should be tailored to your organization's goals and business outcomes. For help with building your business case, see [Create your business case](#) and [Strategy for partner alignment](#).

Answer questions

After you've shared your strategy and it's likely that the finance team will have questions that you'll need to answer. Some of these questions might be challenging, so it's an effective practice to consider what you might be asked and prepare in advance to answer those questions.

Common goals, drivers, and expected outcomes

Consider the common goals, drivers, and expected outcomes of a Chief Technology Officer (CTO), Chief Information Officer (CIO), Chief Financial Officer (CFO), and the finance team:

	CTO/CIO view	CFO view
Flexibility	Cloud enables dynamic reallocation of IT resources from lower value IT projects to high value projects.	Cloud enables flexible budgetary allocation
Resiliency	More fault-tolerant options for backup, disaster recovery, and security	Lower required cashflow to support disaster recovery and backup
Agility	Higher capability of new technology adoption	New technology adoption without new budgetary allocation
Strategic value	Change IT from a cost center to a profit center. IT personnel evolve into consultants to business units, providing means to innovation, and time to market.	IT as an asset for CFO. New reusable processes, toolsets, scripts, compute images, and ways to increase business agility and innovation
Efficiency	More rapid service delivery to the business units, with focus on driving strategic value.	More granular activity-based costing (ABC), as on-premises cost tracking per workload may be fuzzy. Easier to target the costs per workload or application in the cloud.
Financial benefits	Cloud enables doing more with less, as budgets given to CTO and CIOs are typically constant or shrinking.	Reduced cost. Enables reinvestment opportunities into higher value projects.

Next steps

Learn more about how the cloud can advance your business strategy.

[Using cloud economics to advance your business strategy](#)

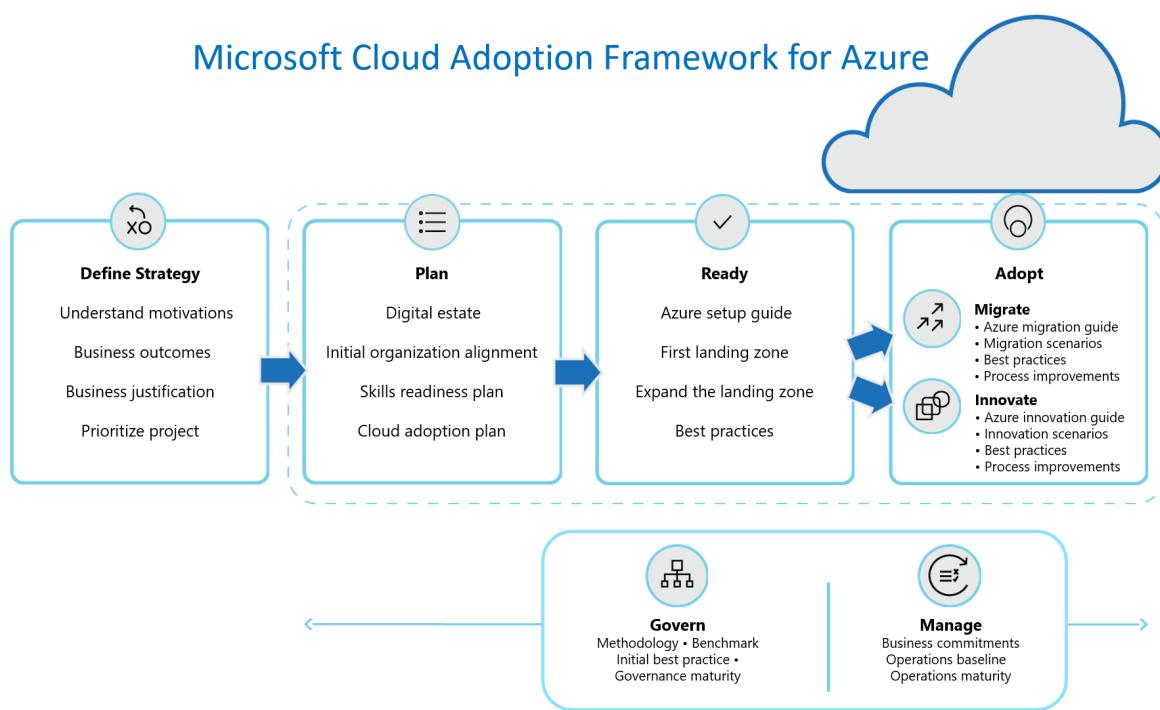
Balance competing priorities

Article • 02/28/2023

The success of any digital transformation is determined by the business and technology stakeholders' ability to balance competing priorities.

Like other digital transformations, cloud adoption exposes competing priorities throughout the adoption lifecycle. And as with other forms of transformation, the ability to balance those priorities has a significant effect on the realization of business value. Balancing these competing priorities requires open and sometimes difficult conversations among stakeholders, and sometimes with individual contributors.

This article outlines some of the competing priorities commonly discussed during the implementation of each methodology. It can help you prepare for those discussions when you develop your cloud adoption strategy.



The following sections align to the flow in the preceding cloud adoption lifecycle diagram. However, it's important to recognize that cloud adoption is an iterative process, not a sequential one, and that these competing priorities might reemerge at various points during your cloud adoption journey.

The general theme of the Cloud Adoption Framework approach

Monolithic solutions and advanced planning are both built on a series of assumptions that might prove to be inaccurate over time. Adopting the cloud is often a new experience for a business and its technical teams. As with most new experiences, there's some likelihood that those initial assumptions will be proven incorrect.

Following proven agile principles of delayed technical decisions is the favored approach for most guidance within this framework. That approach follows a consistent pattern: establish a general end state, move quickly to initial implementation, test and validate assumptions, and refactor early to address faulty assumptions. This type of growth mindset maximizes learning and minimizes risk to business value, but it requires some comfort with ambiguity.

Ambiguity can sometimes be scarier, or more dangerous, than false assumptions. Although this framework prioritizes learning and addressing ambiguity during implementation, many situations require the team to prioritize analysis-based or assumption-based approaches. The following sections provide at least one "expanded scope example" to illustrate when a second deeper iteration might be valuable.

Balance during the strategy phase

The core objective of the Strategy methodology is to develop alignment among stakeholders. After it's defined, the aligned strategic position drives behavior throughout each of the methodologies to ensure that technical decisions align with desired business outcomes. Fostering alignment among stakeholders creates a common set of competing priorities: *depth of justification* versus *time to business impact*.

Competing priorities:

- **Depth of justification:** Stakeholders often want a deep financial analysis and full business justification before they're comfortable with aligning to a strategic direction. Unfortunately, that level of analysis might require an extended time period to allow for data collection and analysis.
- **Time to business impact:** Conversely, stakeholders are often held accountable for delivering business outcomes within defined timeframes. Time-consuming analysis and assessment can put those outcomes at risk before the technical work even begins.

Minimum scope: Finding the correct balance requires stakeholder discussions early in the process. The Strategy methodology suggests limiting the scope of alignment during this early effort. In the suggested approach, stakeholders focus on aligning around a set of core motivations, measurable outcomes, and a high-level business justification.

Stakeholders should then quickly commit to a few initial projects or pilots to drive required learning opportunities.

Expanded scope example: If the initial business analysis indicates a high risk of negatively affecting the business, stakeholders might need to slow down and more cautiously evaluate a deeper analysis during business justification.

Balance during the planning phase

As during the strategy phase, there's a need during the planning phase to balance the depth of initial planning versus delayed technical decisions.

Competing priorities:

- **Depth of initial planning** for technical implementation in the cloud is often based on many assumptions. Especially when there are skill gaps on the team, the environment suffers from discovery gaps, or the workloads don't have clearly defined architectural end states. All of these assumptions are common in detailed cloud adoption plans. Experimentation, pilots, and qualitative analysis are required to verify or improve upon these assumptions.
- **Delayed technical decisions** are based on the assumption that the later a technical decision is made, the more accurate it is. Following principles of agile product planning helps to delay technical decisions, allowing them to happen at the right time, based on sufficient information. However, this approach results in more ambiguity in the initial plan.

Minimum scope: We suggest agile product development approaches to drive prompt action within manageable plans. The Plan methodology recommends the following steps to achieve this balance:

1. Inventory the full digital estate by using automated discovery tools, but use incremental rationalization to plan as far as the next 1 to 3 months of work.
2. Ensure proper organizational alignment to move quickly.
3. Create a skills readiness plan for the assigned team. Use the strategy and plan template to quickly deploy an initial backlog.

Expanded scope example: The delivery of a cloud adoption plan might sometimes be in response to a time-sensitive or high-impact business event. When success requires the movement of a high number of assets during a fixed period of time, the preceding steps are often followed by a deeper planning effort. The key to success in these scenarios is to plan enough to get going, and then later plan for the full engagement. This approach reduces the likelihood that planning will block business outcomes.

Balance during the readiness phase

When teams are preparing for their first steps into cloud adoption, there are often competing priorities between time to adoption and long-term operations. The team might struggle with the balance between being well suited to deliver on the task at hand and being well managed. This struggle is necessary in traditional IT environments, where the act of developing a platform requires physical assets and acquisition cycles. However, when the entire IT platform is defined in code, traditional development tactics, like refactoring, reduce the need to be well managed from the beginning.

Competing priorities:

- **Long-term operations:** Organizations are often blocked by the desire to have a cloud environment that meets feature parity with current operations management, governance, and security systems. In one study, more than 90 percent of organizations needed support to get past that mindset. This blocker can create months of delay, slowing down or preventing business impact.
- **Time to adoption:** Cloud-based tools like Azure Policy, Azure Blueprints, and management groups can simplify the process of refactoring across the IT platform. Additionally, predefined landing zones provide recommendations to accelerate deployment toward an environment that already meets many feature parity requirements. These tools provide opportunities to accelerate time to market with minimal effect on long-term operations.

Minimum scope: The Ready methodology outlines a direct path from rapid adoption to long-term operations. This approach starts with a basic introduction to the tools that you can use to refactor your environment. The tools take into account your requirements and guide you to a selection of predefined landing zones, each delivered with infrastructure as code models. You can then refactor the code during the course of cloud adoption to improve operations, security, and management.

Expanded scope example: For teams whose adoption plan calls for a mid-term objective (within 24 months) to host more than 1,000 assets (applications, infrastructure, or data assets) in the cloud, we recommend a more robust view of landing zones. In these situations, you should consider the Govern and Manage methodologies during your initial landing zone conversations. However, this deeper consideration often adds weeks or months to a cloud adoption plan. To minimize the effect on business outcomes, the adoption team should pilot actual workloads in the cloud while, at the same time, creating a more mature landing zone and central architecture solution.

Balance during the migration phase

During migration, adoption teams commonly assume that workloads will be rehosted in the cloud in their current configuration. This assumption competes with a forward-looking plan to rearchitect every workload to better take advantage of cloud capabilities. The two views aren't mutually exclusive and can be complimentary when you manage them by using a common process.

Competing priorities:

- **Rehost:** Organizations often equate migration with a *lift-and-shift* approach that replicates all assets to the cloud in their current configuration. This approach results in little drift within the IT portfolio. It's also the fastest way to retire assets in an existing datacenter.
- **Rearchitect:** Modernizing the architecture of each workload maximizes the value of cloud adoption across cost, performance, and operations. However, this approach is slower and often requires access to each application's source code.

Minimum scope: During early-stage planning, use the rehost option for planning, with a clear understanding that this choice is based on an initial business assumption and isn't a technical decision. In the Migrate methodology, the cloud adoption team then challenges this assumption for each migrated workload. This methodology follows the assess/migrate/promote approach for each workload or group of workloads to create a migration factory. During the assessment phase, the adoption team evaluates the technical fit and architecture of each workload. That assessment effort seldom results in a pure lift-and-shift approach because many of the components in the architecture tend to be selected for refactoring and modernization.

Expanded scope example: For mission-critical or high-sensitivity workloads, like mainframe or multtier microservices applications, a more thorough assessment of the workload might be required during the assessment phase. In these rearchitecture situations, you should use the Azure Well-Architected Review and the [Azure Well-Architected Framework](#) to refine workload requirements during the assessment.

Balance during the innovation phase

True customer-facing innovation frequently creates conflicting priorities between the need to deliver on a planned feature set and implementing a customer-empathy development process.

Competing priorities:

- **Feature focus:** Initial plans for innovation build on the existing digital estate and cloud capabilities to deliver a set of features that meet a customer need. It's easy

to allow the plan to drive technical implementation, leading to a feature-focused development effort. This approach often leads to temporary stakeholder satisfaction but reduces the likelihood of driving innovation that influences customer behavior.

- **Customer empathy:** Initial plans are an important part of the business side of development and should be included in regular reporting. However, learning, measuring, and building with customer empathy as a goal is a more accurate measure of success in an innovation effort. Focusing on customers over features is more likely to result in short-term and long-term customer satisfaction and business impact.

Minimum scope: The Innovate methodology illustrates how to integrate strategy and plans through business-value consensus. The guide then introduces cloud-native tools that can accelerate each discipline of innovation and best practices for implementation. Finally, the process improvements section demonstrates approaches for building customer empathy while respecting plans and strategies across the cloud adoption journey. This approach focuses on delivering innovation with the use of as little technology as possible.

Expanded scope example: An innovation is sometimes dependent on mission-critical or high-sensitivity workloads. When the customer is an internal user, the development effort might be both mission-critical and high-sensitivity during the earliest iterations. In these scenarios, adoption teams should use the Azure Well-Architected Review and the [Azure Well-Architected Framework](#) to evaluate advanced architectural design early in the process.

Balance during the governance phase

The practice of cloud governance is a balance between two competing priorities: speed and agility versus a well-governed environment. The cloud governance team focuses on evaluating and minimizing risks to the business through uniform controls and minimizing change. The adoption team focuses on driving business outcomes, which require new risks and inherently creates change.

Competing priorities:

- **Well-governed:** Every control designed to minimize risk blocks some aspect of change or limits design options. Control is essential to a well-governed environment. However, when controls are designed and deployed in isolation, they can be as damaging as the risks they're intended to prevent.
- **Speed and agility:** Speed and agility are fundamental business requirements in the digital economy. Both require the ability to drive change with minimal blockers to

innovation or adoption. But when change is driven without governance, it generates new risks that could harm the business in unexpected ways.

Minimum scope: The Govern methodology suggests that neither governance nor adoption should ever happen in isolation. This methodology starts with an understanding of the governance disciplines and a conversation about business risk, policy, and process. As an active participant throughout cloud adoption, the governance team can implement a minimum set of safeguards to address the tangible risks within the cloud adoption plan. Over time, the governance team can refactor and expand those safeguards to meet new risks. This approach maximizes learning and innovation while minimizing risk.

Expanded scope example: When business risk is high, especially early in adoption, the cloud governance team might need to speed up the expansion of governance implementations. You can use the same guidance and exercises to add this higher level of governance, but you might need to go faster. In some scenarios, an advanced state of governance might even be required while you're developing the first landing zones.

Balance during the management phase

The IT business model for operations management has continuously evolved over the past decade. As hardware maintenance moves further from IT's core value proposition, the view on operations management has shifted. As IT increases a focus on delivering business value, operations management teams are conflicted by the need to balance no-ops and low-ops versus broad investments.

Competing priorities:

- **Broad investments:** Investing equally in outage avoidance, rapid recovery, and monitoring across the environment is the traditional approach to operations management. This approach can be expensive and sometimes duplicates the supporting products provided by the cloud vendor.
- **No-ops and low-ops:** Use cloud-native operations tools to minimize repetitive and recurring tasks that were previously delivered by your employees. Reducing these operational dependencies frees employees to drive value in other ways. In isolation, however, this approach can lead to subpar operations support.

Minimum scope: The Manage methodology suggests establishing a cloud-native, no-ops baseline. Acknowledging that the no-ops baseline won't meet all business needs, work with the business to define commitments and better align investments. Expand the baseline to meet common needs for all workloads. Then enable platform teams or

specific workload teams to maintain well-managed solutions within a well-managed environment.

Expanded scope example: In most environments, the business value of a small percentage of workloads justifies deep investments in operations from IT. For those workloads, the IT team might want to use the Azure Well-Architected Review and the Azure Well-Architected Framework to guide deeper operations.

Balance during the organization phase

The competing priorities described in this article reflect IT's drive to deliver on business demands for speed and agility. The same shift is appearing in changes to org charts or virtual team structures to provide better support for business outcomes. As IT leaders reflect on team structures, two competing priorities are commonly addressed: centralized control versus delegated control.

Competing priorities:

- **Centralized control:** This operating model focuses on the centralization of all controls that are required to enforce rigid policies. In this model, IT serves as a blocker to innovation, speed, and agility. However, IT can ensure a higher degree of stability, compliance, and security.
- **Delegated control:** In this distributed operating model, it's assumed that each DevOps team or business application team will provide their own set of controls, based on the solutions that are required to deliver on business objectives. In this model, IT provides guidelines to help teams avoid risks but minimizes the number of enforced technical constraints whenever possible.

Minimum scope: Most organizations go through a natural set of evolutions over time. The Organize methodology outlines the most common series of evolutions. We recommend that teams try to move toward a cloud center of excellence (CCoE) structure to deliver delegated control approaches.

Expanded scope example: Many situations trigger a need for centralized control. Third-party compliance requirements and temporary security exposure are two examples of triggers for centralized control. In these situations, there's commonly a need to establish limiting policies and rigid, fixed controls. However, to enable innovation and adoption to continue, we encourage central IT teams to deliver those controls based on the criticality and sensitivity of each workload. Providing environments with less control but a reduced scope or risk profile allows flexibility even when control is required.

Next steps

Learn to [balance migration, innovation, and experimentation](#) to maximize the value your cloud migration efforts.

[Balance the portfolio](#)

Balance the portfolio

Article • 02/28/2023

Cloud adoption is a portfolio-management effort, cleverly disguised as technical implementation. Like any portfolio management exercise, balancing the portfolio is critical. At a strategic level, this means balancing migration, innovation, and experimentation to get the most out of the cloud. When the cloud adoption effort leans too far in one direction, complexity finds its way into the adoption efforts. This article will guide the reader through approaches to achieve balance in the portfolio.

General scope expansion

Balancing the portfolio is strategic in nature. As such, the approach taken in this article is equally strategic. To ground the strategy in data-driven decisions, this article assumes the reader has evaluated the existing [digital estate](#) or has begun that process. The objective of this approach is to aid in evaluating workloads to ensure proper balance across the portfolio through qualitative questions and portfolio refinement.

Document business outcomes

Before balancing the portfolio, it is important to document and share the business outcomes driving the cloud-migration effort. The following table can help document and share desired business outcomes. It's important to note that most businesses are pursuing several outcomes at a time. The importance of this exercise is to clarify the outcomes that are most directly related to the cloud migration effort:

Outcome	Measured by	Goal	Time frame	Priority for this effort
Reduce IT costs	Datacenter budget	Reduce by \$2M USD	12 months	#1
Datacenter exit	Exit from datacenters	2 datacenters	6 months	#2
Increase business agility	Improve time to market	Reduce deployment time by six months	2 years	#3
Improve customer experience	Customer satisfaction (CSAT)	10% improvement	12 months	#4

 **Important**

The above table is a fictional example and should not be used to set priorities. In many cases, this table could be considered an antipattern by placing cost savings above customer experiences.

The above table could accurately represent the priorities of the cloud strategy team and the cloud adoption team. Due to short-term constraints, this team is placing a higher emphasis on IT cost reduction and prioritizing a datacenter exit as a means to achieve the desired IT cost reductions. However, by documenting the competing priorities in this table, the cloud adoption team is empowered to help the cloud strategy team identify opportunities to better align implementation of the overarching portfolio strategy.

Move fast while maintaining balance

The guidance regarding [incremental rationalization of the digital estate](#) suggests an approach in which the rationalization starts with an unbalanced position. The cloud strategy team should evaluate every workload for compatibility with a rehost approach. Such an approach is suggested because it allows for the rapid evaluation of a complex digital estate based on quantitative data. Making such an initial assumption allows the cloud adoption team to engage quickly, reducing time to business outcomes. However, as stated in that article, qualitative questions will provide the necessary balance in the portfolio. This article documents the process for creating the promised balance.

Importance of sunset and retire decisions

The table in the [documenting business outcomes](#) section above misses a key outcome that would support the number one objective of reducing IT costs. When IT costs reductions rank anywhere in the list of business outcomes, it is important to consider the potential to sunset or retire workloads. In some scenarios, cost savings can come from not migrating workloads that don't warrant a short-term investment. Some customers have reported cost savings in excess of 20% total cost reductions by retiring underutilized workloads.

To balance the portfolio, better reflecting sunset and retire decisions, the cloud strategy team and the cloud adoption team are encouraged to ask the following questions of each workload during assessment and migration phases:

- Has the workload been used by end users in the past six months?
- Is end-user traffic consistent or growing?
- Will this workload be required by the business 12 months from now?

If the answer to any of these questions is "no", then the workload could be a candidate for retirement. If retirement potential is confirmed with the application owner, then it may not make sense to migrate the workload. This prompts for a few qualification questions:

- Can a retirement plan or sunset plan be established for this workload?
- Can this workload be retired prior to the datacenter exit?

If the answer to both of these questions is "yes", then it would be wise to consider **not** migrating the workload. This approach would help meet the objectives of reducing costs and exiting the datacenter.

If the answer to either question is "no", it may be wise to establish a plan for hosting the workload until it can be retired. This plan could include moving the assets to a lower-cost datacenter or alternative datacenter, which would also accomplish the objectives of reducing costs and exiting one datacenter.

Adopt process changes

Balancing the portfolio requires additional qualitative analysis during the Adopt phase, which will help drive simple portfolio rationalization.

Based on the data from the table in the [documenting business outcomes](#) section above, there is a likely risk of the portfolio leaning too far into a migration-focused execution model. If customer experience was top priority, an innovation heavy portfolio would be more likely. Neither is right or wrong, but leaning too far in one direction commonly results in diminishing returns, adds unnecessary complexity, and increases execution time related to cloud adoption efforts.

To reduce complexity, you should follow a traditional approach to portfolio rationalization, but in an iterative model. The following steps outline a qualitative model to such an approach:

- The cloud strategy team maintains a prioritized backlog of workloads to be migrated.
- The cloud strategy team and the cloud adoption team host a release planning meeting prior to the completion of each release.
- In the release planning meeting, the teams agree on the top 5 to 10 workloads in the prioritized backlog.
- Outside of the release planning meeting, the cloud adoption team asks the following questions of application owners and subject matter experts:

- Could this application be replaced with a platform as a service (PaaS) equivalent?
- Is this application a third-party application?
- Has budget been approved to invest in ongoing development of the application in the next 12 months?
- Would additional development of this application improve the customer experience? Create a competitive differentiator? Drive additional revenue for the business?
- Will the data within this workload contribute to a downstream innovation related to BI, machine learning, IoT, or related technologies?
- Is the workload compatible with modern application platforms like Azure App Service?
- The answers to the above questions and any other required qualitative analysis would then influence adjustments to the prioritized backlog. These adjustments may include:
 - If a workload could be replaced with a PaaS solution, it may be removed from the migration backlog entirely. At a minimum, additional due diligence to decide between rehost and replace would be added as a task, temporarily reducing that workload's priority from the migration backlog.
 - If a workload is (or should be) undergoing development advancement, then it may best fit into a refactor-rearchitect-rebuild model. Since innovation and migration require different technical skills, applications that align to a refactor-rearchitect-rebuild approach should be managed through an innovation backlog rather than a migration backlog.
 - If a workload is part of a downstream innovation, then it may make sense to refactor the data platform, but leave the application layers as a rehost candidate. Minor refactoring of a workload's data platform can often be addressed in a migration or an innovation backlog. This rationalization outcome may result in more detailed work items in the backlog, but otherwise no change to priorities.
 - If a workload isn't strategic but is compatible with modern, cloud-based application hosting platforms, then it may be wise to perform minor refactoring on the application to deploy it as a modern application. This can contribute to the overall savings by reducing the overall IaaS and OS licensing requirements of the cloud migration.
 - If a workload is a third-party application and that workload's data isn't planned for use in a downstream innovation, then it may be best to leave as a rehost option on the backlog.

These questions shouldn't be the extent of the qualitative analysis completed for each workload, but they help guide a conversation about addressing the complexity of an

imbalanced portfolio.

Migration process changes

During migration, portfolio balancing activities can have a negative impact on migration velocity (the speed at which assets are migrated). The following guidance will expand on why and how to align work to avoid interruptions to the migration effort.

Portfolio rationalization requires diversity of technical effort. It is tempting for cloud adoption teams to match that portfolio diversity within migration efforts. Business stakeholders often ask for a single cloud adoption team to address the entire migration backlog. This is seldom an advisable approach, in many cases this can be counterproductive.

These diverse efforts should be segmented across two or more cloud adoption teams. Using a two-team model as an example mode of execution, team 1 is the migration team and team 2 is the innovation team. For larger efforts, these teams could be further segmented to address other approaches like replace/PaaS efforts or minor refactoring. The following outlines the skills and roles needed to rehost, refactor, or minor refactor:

Rehost: Rehost requires team members to implement infrastructure focused changes. Generally using a tool like Azure Site Recovery to migrate VMs or other assets to Azure. This work aligns well to datacenter admins or IT implementors. The cloud migration team is well structured to deliver this work at high scale. This is the fastest approach to migrate existing assets in most scenarios.

Refactor: Refactor requires team members to modify source code, change the architecture of an application, or adopt new cloud services. Generally this effort would use development tools like Visual Studio and deployment pipeline tools like Azure DevOps to redeploy modernized applications to Azure. This work aligns well to application development roles or DevOps pipeline development roles. The cloud innovation team is best structured to deliver this work. It can take longer to replace existing assets with cloud assets in this approach, but the applications can take advantage of cloud-native features.

Minor refactoring: Some applications can be modernized with minor refactoring at the data or application level. This work requires team members to deploy data to cloud-based data platforms or to make minor configuration changes to the application. This may require limited support for data or application development subject matter experts. However, this work is similar to the work conducted by IT implementors when deploying third-party applications. This work could easily align with the cloud migration team or

the cloud strategy team. While this effort is not nearly as fast as a rehost migration, it takes less time to execute than refactor efforts.

During migration, efforts should be segmented in the three ways listed above and executed by the appropriate team in the appropriate iteration. While you should diversify the portfolio, also ensure that efforts stay very focused and segregated.

Next steps

Understand how [global market decisions](#) can affect your transformation journey.

[Understand global markets](#)

How will global market decisions affect the transformation journey?

Article • 01/09/2024

The cloud opens new opportunities to perform on a global scale. Barriers to global operations are significantly reduced, by empowering companies to deploy assets in their markets, without the need to invest heavily in new datacenters. Unfortunately, this also adds a great deal of complexity from technical and legal perspectives.

Data sovereignty

Many geopolitical regions have data sovereignty regulations. The regulations restrict where data can be stored, what data can leave the country/region of origin, and what data can be collected that's about citizens of that country/region. Before you operate a cloud-based solution in a foreign geography, you should understand how that cloud provider handles data sovereignty.

Find more information about:

- [Azure approaches for each country/region](#).
- [Compliance in Azure](#).
- [Sovereign strategies](#).

The remainder of this article assumes legal counsel has reviewed and approved operations in a foreign geopolitical region.

Business units

It is important to understand which business units operate in foreign locales and which countries/regions are affected. This information will be used to design solutions for hosting, billing, and deployments to the cloud provider.

Employee usage patterns

It is important to understand how global users access applications that are not hosted in the same country/region as the user. Global wide-area networks (WANs) route users based on existing networking agreements. In a traditional on-premises world, some constraints limit WAN design. Those constraints can lead to poor user experiences if not properly understood before cloud adoption.

In a cloud model, commodity internet opens up many new options as well. Communicating the spread of employees across multiple geographies can help the cloud adoption team design WAN solutions that create positive user experiences **and** potential reduce networking costs.

External user usage patterns

It is equally important to understand the usage patterns of external users, like customers or partners. Much like employee usage patterns, external user usage patterns can negatively affect performance of cloud deployments. When a large or mission-critical user base resides in a foreign country/region, it could be wise to include a global deployment strategy into the overall solution design.

Next steps

Learn about the [skills needed during the strategizing phase](#) of your cloud adoption journey.

[Skills needed during the strategizing phase](#)

Define a sovereignty strategy

Article • 01/09/2024

This article describes how to plan your sovereignty strategy when you use cloud services. Many geopolitical regions have regulations for handling specific types of data, such as privacy-sensitive data and government data. The regulations typically enforce sovereignty requirements related to data residency, the control over data, and sometimes operational independence (referred to as *autarky*).

When your organization needs to adhere to these regulations, you should define a strategy to meet the sovereignty requirements. If your organization shifts from on-premises services to cloud services, you must adjust the sovereignty strategy accordingly.

Modernize your sovereignty strategy

For your on-premises datacenter, you're responsible for most aspects that are typically associated with sovereignty, including:

- Datacenters, where data is stored and processed.
- Access to the datacenters and physical infrastructure.
- Hardware and software, including the hardware and software supply chain.
- Assurance processes that validate hardware and software.
- Infrastructure and processes that ensure business continuity if there's a disaster or geopolitical event.
- Configurations and processes that determine who has access to which data and systems.
- Tools and processes that secure data and systems against outside and inside threats.

When you adopt cloud services, the responsibility of these aspects shifts to a [shared responsibility](#). Your compliance team changes the strategy that they use to determine if sovereignty requirements are met. The compliance team considers:

- The compliance of the cloud services. How do the cloud provider's services meet sovereignty and compliance requirements?
- The compliance of the systems and processes that your organization is responsible for. Which tools are available to help you meet sovereignty and compliance requirements, and how do you use these tools?

The compliance team might need to work with a regulator to get permission to use alternative methods that achieve the same goals. In some cases, a regulation might need to be changed by adding more options or adjusting a directive to use a certain solution to obtain an intended result. Changing regulation can be a lengthy process. However, it might be possible to get exemptions if you can demonstrate you have achieved the intent of a regulation.

For example, a regulation might restrict organizations from using certain cloud services because the isolation requirements can only be met with hardware isolation that's typically not available in the cloud. But the intended result can also be obtained with virtual isolation. As part of your strategy, you need to determine how to work with regulators and auditors when these potential blockers arise.

For more information about how to meet your compliance and sovereignty needs, see [Microsoft Cloud for Sovereignty](#).

Compliance of cloud services

The compliance team uses various sources and methods to verify cloud service compliance, including:

- **Vendor documentation** about how their services work and how to use them, for example the US Federal Risk and Authorization Management Program ([FedRAMP](#)) product documentation and system security plans.
- **Independent auditor certifications** that certify compliance to global, regional, and industry compliance frameworks. For more information, see [Compliance offerings for Microsoft 365, Azure, and other Microsoft services](#).
- **Audit reports** that independent auditors create to provide insights into how cloud services meet the requirements of global, regional, and industry compliance frameworks. Some audit reports are available in the [Service Trust Portal](#).
- **Audits** that are performed by or on behalf of the compliance team via vendor audit offerings, such as the [Government Security Program](#) (available to select customers only).
- **Transparency logs** that provide details about when Microsoft engineers access your resources.

The combination of sources and methods that your compliance team uses depends on the level of insight that you need, the trust you have in the different options, and your resources and budget. A third-party auditor certification eliminates the need for your

team to perform an audit and costs less but requires trust in the auditor and audit process.

Compliance of your systems and processes

Your organization's compliance processes and systems can benefit from the added capabilities of cloud services. You can use these capabilities to:

- Enforce or report on technical policies. For example, you might block the deployment of services or configurations or report on violations that don't meet technical requirements for sovereignty and compliance.
- Use pre-built policy definitions that are aligned to specific compliance frameworks.
- Log and monitor audits.
- Use security tools. For more information, see [Define a security strategy](#).
- Perform technical assurance and monitoring capabilities, such as [Azure confidential computing](#).

Carefully consider these capabilities for your organization's environment and individual workloads. For each capability, consider the amount of effort needed, the applicability, and the function. For example, policy enforcement is a relatively simple method that supports compliance, but it can restrict which services you can use and how you can use them. In comparison, technical assurance takes considerable effort and is more restrictive because it's only available for a few services. It also requires a significant amount of knowledge.

Adopt shared responsibility

When you adopt cloud services, you adopt a shared responsibility model. Determine which responsibilities shift to the cloud provider and which remain with you. Understand how those changes affect the sovereignty requirements for regulations. For more information, see the resources in [Compliance of cloud services](#). To get a high-level view, consider the following resources:

- [Azure infrastructure security](#) describes how Microsoft provides protection for the physical infrastructure.
- [Azure platform integrity and security](#) describes how Microsoft provides protection against threats to the platform and the technical assurance processes.

- [Data residency in Azure](#) describes data residency features. For customers in the European Union (EU), see [Microsoft EU Data Boundary](#).

The cloud provider partially provides business continuity via the resilience of the platform by ensuring the continuity of critical systems that operate the cloud. The services that a workload uses provide continuity options that you can use to build your workloads. Or you can use other services, such as [Azure Backup](#) or [Azure Site Recovery](#). For more information, see [Azure reliability documentation](#).

The cloud provider is responsible for securing access to the cloud platform from both internal and external threats. Customers are responsible for configuring their systems to secure their data through identity and access management, encryption, and other security measures. For more information, see [Define a security strategy](#).

Use classifications to differentiate data

Different types of data and workloads can have different sovereignty requirements, depending on factors such as the confidentiality of the data and whether it contains privacy-sensitive data. It's important to understand which data classifications apply to your organization and which data and systems are subject to which classifications. Some data and applications are subject to multiple regulations, which can create the need for combined requirements. For example, there might be a regulation related to the confidentiality of data and the criticality of a system. The resulting classifications might be high confidentiality and low criticality or medium confidentiality and high criticality.

When you comply with sovereignty requirements, it can affect other factors, such as cost, resilience, scalability, security, and service richness. For your sovereignty strategy, it's important to apply the right controls to a data classification. A one-size-fits-all approach leads to an environment that favors the highest compliance requirements, which is likely the costliest and least beneficial.

Next steps

- [Cloud for Sovereignty](#) provides insights into sovereign capabilities on the Azure platform and describes how to address sovereignty requirements.
- Security and sovereignty aren't the same, but you can't be sovereign if you aren't secure. You must therefore [define a security strategy](#) that integrates with your sovereignty strategy.

Define a security strategy

Article • 02/28/2023

The ultimate objectives for a security organization don't change with adoption of cloud services, but how those objectives are achieved will change. Security teams must still focus on reducing business risk from attacks and work to get confidentiality, integrity, and availability assurances built into all information systems and data.

Modernize your security strategy

Security teams need to modernize strategies, architectures, and technology as the organization adopts cloud and operates it over time. While the size and number of changes can initially seem daunting, the modernization of the security program allows security to shed some painful burdens associated with legacy approaches. An organization can temporarily operate with legacy strategy and tooling, but this approach is difficult to sustain with the pace of change in cloud and the threat environment:

- Security teams are likely to be left out of cloud adoption decision making if they take a legacy mindset of "arms-length" security where the answer always starts with "no" (instead of working together with IT and business teams to reduce risk while enabling the business).
- Security teams will have a difficult time detecting and defending against cloud attacks if they use only legacy on-premises tooling and exclusively adhere to network perimeter-only doctrine for all defenses and monitoring.

Monitor and protect at cloud-scale

Defending at cloud-scale is a significant transformation effort, mandating the use of cloud-native detection and automation capabilities, and the introduction of an identity perimeter to help monitor and protect cloud and mobile assets.

- The [Microsoft identity platform](#) helps you incorporate modern authentication and authorization mechanisms into your applications.
- [Microsoft Sentinel](#) provides cloud-native security analytics and threat intelligence across your organization, enabling improved threat detection that makes use of large repositories of threat intelligence, and the nearly unlimited processing and storage capabilities of the cloud.

We recommend that security teams take an agile approach to modernizing security—rapidly modernizing the most critical aspects of the strategy, continuously improving in increments, moving forward.

Security of the cloud and from the cloud

As your organization adopts cloud services, security teams will work toward two main objectives:

- **Security of the cloud** (securing cloud resources): Security should be integrated into the planning and operation of cloud services to ensure that those core security assurances are consistently applied across all resources.
- **Security from the cloud** (using the cloud to transform security): Security should immediately start planning and thinking about how to use cloud technologies to modernize security tools and processes, particularly natively integrated security tools. Security tools are increasingly being hosted in the cloud—providing capabilities that are difficult or impossible to do in an on-premises environment.

Securing software-defined datacenters

Many organizations start by treating cloud resources as another *virtual datacenter*, an effective starting point for security of the cloud. As organizations modernize using security from the cloud, most will find themselves quickly outgrowing this model of thinking. Securing a software-defined datacenter enables capabilities beyond what on-premises models can offer. Cloud-hosted security tools offer:

- Rapid enablement and scaling of security capabilities.
- Highly effective asset inventory and security configuration hygiene discovery.

Deploying [Microsoft Defender for Cloud](#) enables the **continuous assessment of your organization's security posture and controls**. It strengthens the security posture of your cloud resources, and with its integrated Microsoft Defender plans, Defender for Cloud protects workloads running in Azure, hybrid, and other cloud platforms. Learn more about [Microsoft Defender for Cloud](#).

Note

Azure Security Center and Azure Defender are now called Microsoft Defender for Cloud. We've also renamed Azure Defender plans to Microsoft Defender plans. For example, Azure Defender for Storage is now Microsoft Defender for Storage.

Learn more about the recent renaming of Microsoft security services .

The right level of security friction

Security naturally creates friction that slows down processes, it's critical to identifying which elements are healthy in your DevOps and IT processes and which are not:

- **Healthy friction:** Much like the resistance in exercise makes a muscle stronger, integrating the right level of security friction strengthens the system or application by forcing critical thinking at the right time. This typically takes the form of considering how and why an attacker might try to compromise an application or system during design (known as [threat modeling](#)), and reviewing, identifying, and ideally fixing potential vulnerabilities an attacker can exploit in software code, configurations, or operational practices.
- **Unhealthy friction:** Impedes more value than it protects. This often happens when security bugs generated by tools have a high false positive rate (such as false alarms) or when the effort to discover or fix security issues far exceeds the potential impact of an attack.

Standalone and integrated responsibilities

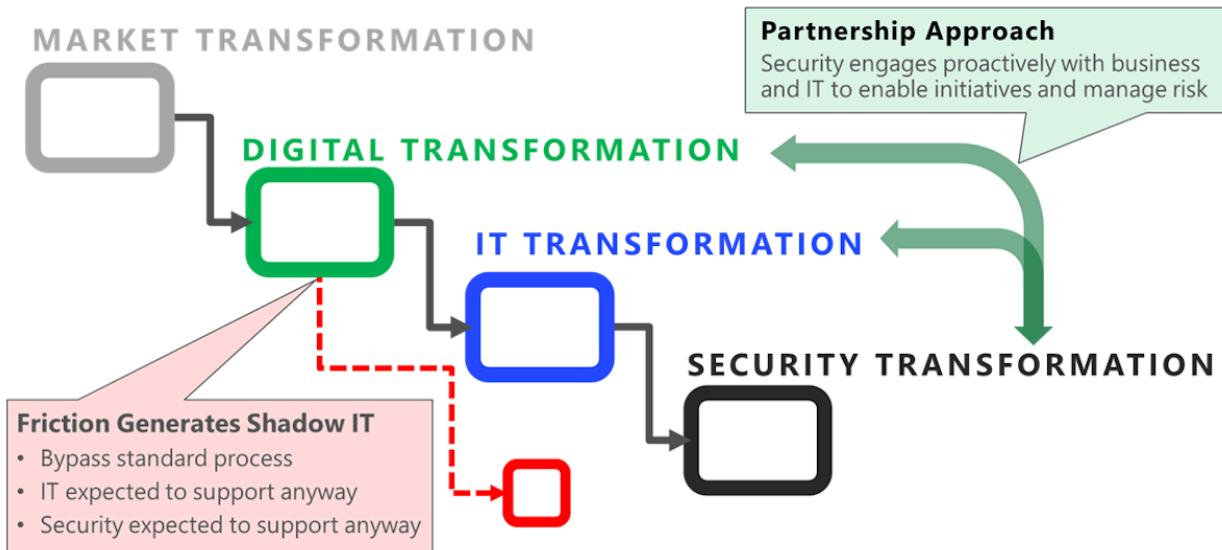
Providing confidentiality, integrity, and availability assurances requires security experts to operate dedicated security functions and work closely with other teams in the organization:

- **Unique security functions:** Security teams perform independent functions that are not found elsewhere in the organization, such as security operations, vulnerability management (such as for [virtual machines](#), [containers](#)), and other functions.
- **Integrating security into other functions:** Security teams also serve as subject matter experts to other teams and functions in the organization who are driving business initiatives, assessing risk, designing or developing applications, and operating IT systems. Security teams advise these teams with expertise and context on attackers, attack methods and trends, vulnerabilities that could allow unauthorized access, and options for mitigation steps or workarounds and their potential benefits or pitfalls. This function of security resembles that of a quality function as it will be woven into many places large and small in support of a single outcome.

Executing on these responsibilities while keeping up with the rapid pace of change in the cloud and the transformation of business requires security teams to modernize their tools, technologies, and processes.

Transformations, mindsets, and expectations

Many organizations are managing a chain of multiple simultaneous transformations in the organization. These internal transformations typically start because nearly all external markets are transforming to meet new customer preferences for mobile and cloud technologies. Organizations often face the competitive threat of new startups and the digital transformation of traditional competitors who can disrupt the market.



The internal transformation process typically includes:

- **Digital transformation** of the business to capture new opportunities and stay competitive against digital native startups.
- **Technology transformation** of the IT organization to support the initiative with cloud services, modernized development practices, and related changes.
- **Security transformation** to both adapt to the cloud and simultaneously address an increasingly sophisticated threat environment.

Internal conflict can be costly

Change creates stress and conflict, which can grind decision making to a halt. This is especially true in security where accountability for security risk is often misplaced on the subject matter experts (security teams), rather than on the owners of the assets (business owners) that are accountable for business outcomes and all other risk types. This misplaced accountability often happens because all stakeholders incorrectly view security as a technical or absolute problem to be solved, rather than a dynamic ongoing risk like corporate espionage and other traditional criminal activities.

During this time of transformation, leadership of all teams must work actively to reduce conflict that can both derail critical projects and incentivize teams to bypass security risk mitigation. Internecine conflict between teams can result in:

- **Increased security risk** such as avoidable security incidents or increased business damage from attacks (particularly when teams get frustrated by security and bypass normal processes or when outdated security approaches are easily bypassed by attackers).
- **Negative impact on the business or mission** such as when business processes aren't enabled or updated fast enough to meet market needs (often when security processes hold up key business initiatives).

It's critical to stay aware of relationship health within and between teams to help them navigate the shifting landscape that could leave valuable team members insecure and unsettled. Patience, empathy, and education on these mindsets and the positive potential of the future will help your teams better navigate this period, driving good security outcomes for the organization.

Leaders can help drive culture changes with concrete proactive steps like:

- Publicly modeling the behavior they expect of their teams.
- Being transparent about the challenges of the changes, including highlighting their own struggles to adapt.
- Regularly reminding teams of the urgency and importance of modernizing and integrating security.

Cybersecurity resilience

Many classic security strategies have been focused solely on preventing attacks, an approach that is insufficient for modern threats. Security teams must ensure their strategy goes beyond this and also enables rapid attack detection, response, and recovery to increase resilience. Organizations must assume that attackers will compromise some resources (sometimes called *assume breach*) and work to ensure that resources and technical designs are balanced between attack prevention and attack management (rather than the typical default approach of only attempting to prevent attacks).

Many organizations are already on this journey because they have been managing the steady rise in volume and sophistication of attacks in recent years. This journey often starts with the first major incident, which can be an emotional event where people lose their prior sense of invulnerability and safety. While not as severe as a loss of life, this event can trigger similar emotions starting with denial and ultimately ending in acceptance. This assumption of "failure" might be difficult for some to accept at first, but it has strong parallels to the well-established "fail-safe" engineering principle and the assumption allows your teams to focus on a better definition of success: resilience.

The functions of the [NIST cybersecurity framework](#) serve as a useful guide on how to balance investments between the complementary activities of identify, protect, detect, respond, and recover in a resilient strategy.

More on cybersecurity resilience and the ultimate goals of cybersecurity controls is discussed in [How do you keep your organization's risk down](#).

How the cloud is changing security

Shifting to the cloud for security is more than a simple technology change, it's a generational shift in technology akin to moving from mainframes to desktops and onto enterprise servers. Successfully navigating this change requires fundamental shifts in expectations and mindset by security teams. Adopting the right mindsets and expectations reduces conflict within your organization and increases the effectiveness of security teams.

While these could be part of any security modernization plan, the rapid pace of change in the cloud makes adopting them an urgent priority.

- **Partnership with shared goals.** In this age of fast paced decisions and constant process evolution, security can no longer adopt an "arms-length" approach to approving or denying changes to the environment. Security teams must partner closely with business and IT teams to establish shared goals around productivity, reliability, and security and work collectively with those partners to achieve them.

This partnership is the ultimate form of "shift left"—the principle of integrating security earlier in the processes to make fixing security issues easier and more effective. This requires a culture change by all involved (security, business, and IT), requiring each to learn the culture and norms of other groups while simultaneously teaching others about their own.

Security teams must:

- **Learn** the business and IT objectives and why each is important and how they are thinking about achieving them as they transform.
- **Share** why security is important in the context of those business goals and risks, what other teams can do to meet security goals, and how they should do it.

While not an easy task, it's essential for sustainably securing the organization and its assets. This partnership will likely result in healthy compromises where only the minimum security, business, and reliability goals might be met initially, but incrementally improve steadily over time.

- **Security is an ongoing risk, not a problem.** You can't "solve" crime. At its core, security is just a risk management discipline, which happens to be focused on malicious actions by humans rather than natural events. Like all risks, security is not a problem that can be fixed by a solution, it's a combination of the likelihood and impact of damage from a negative event, an attack. It's most comparable to traditional corporate espionage and criminal activities where organizations face motivated human attackers who have financial incentive to successfully attack the organization.
- **Success in either productivity or security requires both.** An organization must focus on both security and productivity in today's "innovation or become irrelevant" environment. If the organization is not productive and driving new innovation, it could lose competitiveness in the marketplace that causes it to weaken financially or eventually fail. If the organization is not secure and loses control of assets to attackers, it could lose competitiveness in the marketplace that causes it to weaken financially and eventually fail.
- **Nobody's perfect.** No organization is perfect at adopting the cloud, not even Microsoft. Microsoft's IT and security teams grapple with many of the same challenges that our customers do such as figuring out how to structure programs well, balancing supporting legacy software with supporting cutting edge innovation, and even technology gaps in cloud services. As these teams learn how to better operate and secure the cloud, they are actively sharing their lessons learned via documents like this along with others on the [IT showcase site](#), while continuously providing feedback to our engineering teams and third-party vendors to improve their offerings.

Based on our experience, we recommend that teams are held to a standard of continuous learning and improvement rather than a standard of perfection.

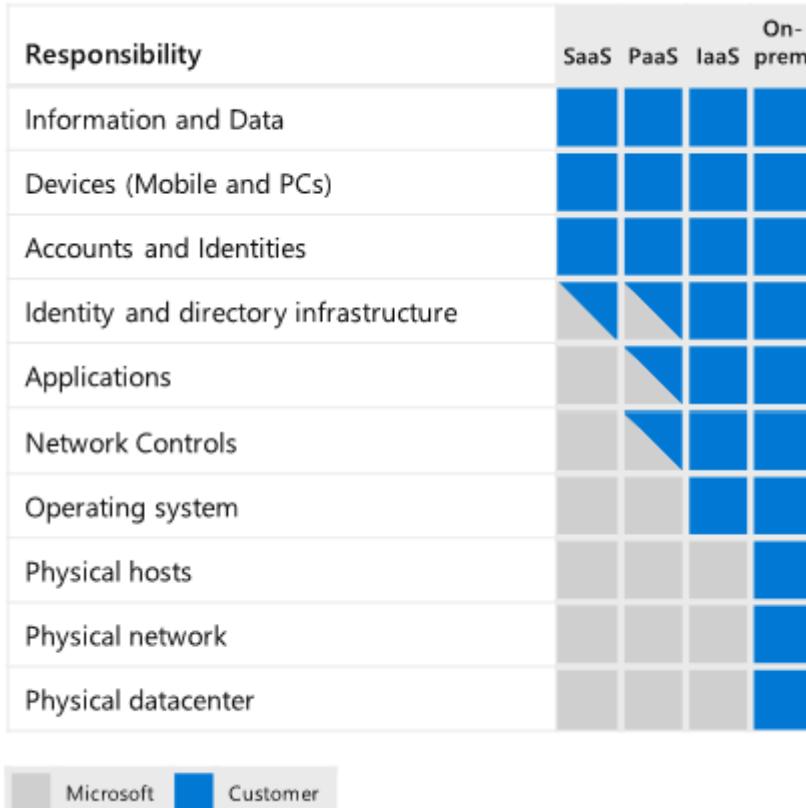
- **Opportunity in transformation.** It's important to view digital transformation as a positive opportunity for security. While it's easy to see the potential downsides and risk of this change, it's easy to miss the massive opportunity to reinvent the role of security and earn a seat at the table where decisions are made. Partnering with the business can result in increased security funding, reduce wasteful repetitive efforts in security, and make working in security more enjoyable as they will be more connected to the organization's mission.

Adopting the shared responsibility model

Hosting IT services in the cloud splits the operational and security responsibilities for workloads between the cloud provider and the customer tenant, creating a de facto

partnership with shared responsibilities. All security teams must study and understand this shared responsibility model to adapt their processes, tools, and skill sets to the new world. This will help avoid inadvertently creating gaps or overlaps in your security posture resulting in security risks or wasted resources.

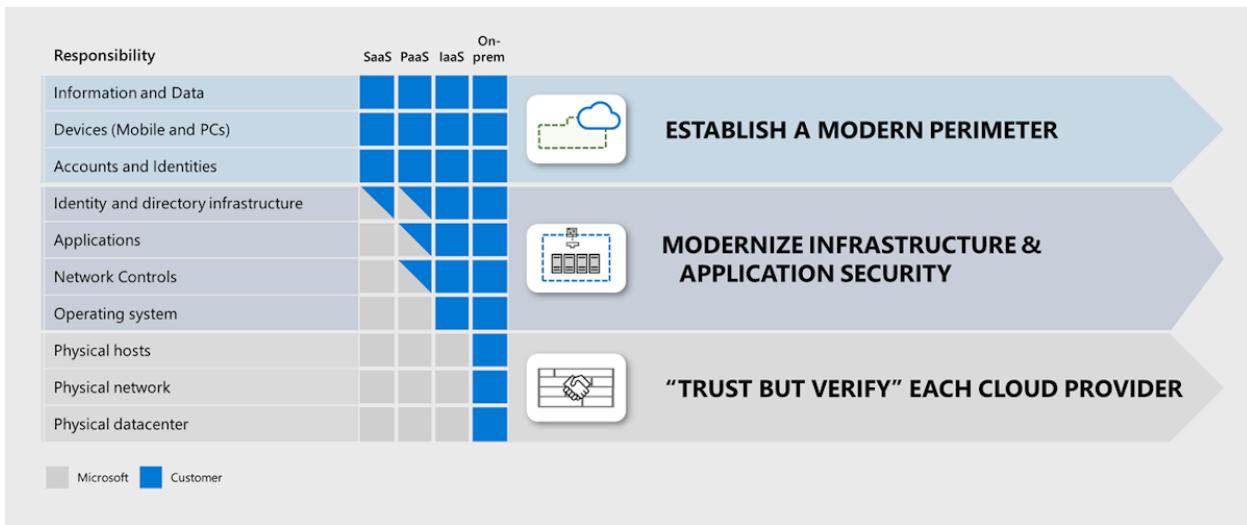
This diagram illustrates how security responsibilities will be distributed between cloud vendors and cloud customer organizations in a de facto partnership:



As there are different models of cloud services, the responsibilities for each workload will vary depending on whether it's hosted on software as a service (SaaS), platform as a service (PaaS), infrastructure as a service (IaaS), or in an on-premises datacenter.

Building security initiatives

This diagram illustrates the three primary security initiatives that most security programs should follow to adjust their security strategy and security program goals for the cloud:



Building a resilient security posture in the cloud requires several parallel complementary approaches:

- **Trust but verify:** For responsibilities performed by the cloud provider, organizations should take a "trust but verify" approach. Organizations should evaluate the security practices of their cloud providers and the security controls they offer to ensure the cloud provider meets the security needs of the organization.
- **Modernize infrastructure and application security:** For technical elements under the organization's control, prioritize modernizing security tooling and associated skill sets to minimize coverage gaps for securing resources in the cloud. This is composed of two different complementary efforts:
 - **Infrastructure security:** Organizations should use the cloud to modernize their approach to protecting and monitoring the common components used by many applications, such as operating systems, networks, and container infrastructure. These cloud capabilities can often include managing infrastructure components across both IaaS and on-premises environments. Optimizing this strategy is important because this infrastructure is a dependency of the applications and data that run on it, which often enable critical business processes and store critical business data.
 - **Application security:** Organizations should also modernize the way they secure the unique applications and technology that is developed by or for their organization. This discipline is changing rapidly with the adoption of agile DevOps processes, the increasing use of open-source components, and introduction of cloud APIs and cloud services to replace application components or interconnect applications.

Getting this right is critical because these applications often enable critical business processes and store critical business data.

- **Modern perimeter:** Organizations should have a comprehensive approach for protecting data across all workloads, organizations should establish a modern perimeter of consistent, centrally managed identity controls to protect their data, devices, and accounts. This is heavily influenced by a zero trust strategy discussed in detail in [Module 3 of the CISO workshop](#).

Security and trust

The use of the word *trust* in security can be confusing. This documentation refers to it in two ways that illustrate useful applications of this concept:

- [Zero trust](#) is a common industry term for a strategic approach to security that assumes a corporate or intranet network is hostile (worthy of *zero trust*) and designs security accordingly.
- [Trust but verify](#) is an expression that captures the essence of two different organizations working together toward a common goal despite having some other potentially divergent interests. This concisely captures many of the nuances of the early stages of partnering with a commercial cloud provider for organizations.

A cloud provider and its practices and processes can be accountable to meet contractual and regulatory requirements and could earn or lose trust. A network is a nonliving connection that cannot face consequences if it's used by attackers (much like you cannot hold a road or a car accountable for criminals using them).

How cloud is changing security relationships and responsibilities

As with previous transitions to a new generation of technology like desktop computing and enterprise server computing, the shift to cloud computing is disrupting long-established relationships, roles, responsibilities, and skill sets. The job descriptions we have become accustomed to over the last few decades do not cleanly map to an enterprise that now includes cloud capabilities. As the industry collectively works to normalize a new model, organizations will have to focus on providing as much clarity as possible to help manage the uncertainty of ambiguity during this period of change.

Security teams are affected by these changes in the business and technology they support as well as their own internal modernization efforts to better orient to threat actors. Attackers are actively evolving to constantly search for the easiest weak points to

exploit in the people, process, and technology of the organization and security must develop capabilities and skills to address these angles.

This section describes the key relationships that frequently change on the journey to the cloud, including lessons learned on minimizing risk and embracing the opportunities to improve:

- **Between security and business stakeholders:** Security leadership will need to increasingly partner with business leaders to enable organizations to reduce risk. Security leaders should support business decision making as security subject matter expert (SMEs) and should strive to grow into trusted advisors to these business leaders. This relationship will help ensure business leaders consider security risks while making business decisions, inform security of business priorities, and help ensure security investments are prioritized appropriately alongside other investments.
- **Between security leadership and team members:** Security leadership should take these insights from business leadership back to their teams to guide their investment priorities.

By setting a tone of cooperation with business leaders and their teams rather than a classic "arms-length" relationship, security leaders can avoid an adversarial dynamic that impedes both security and productivity goals.

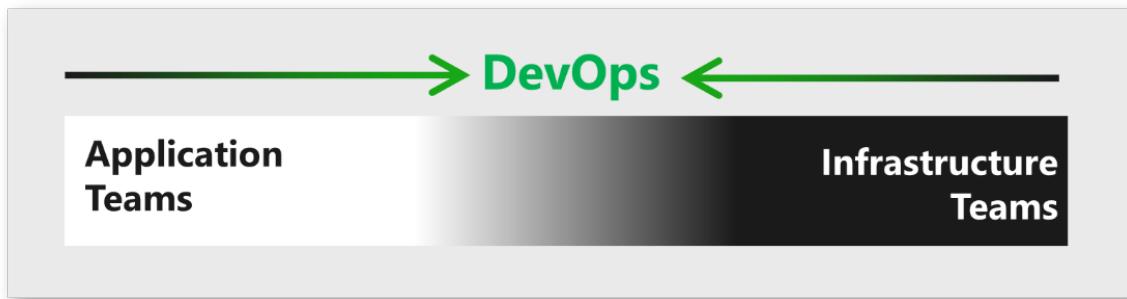
Security leaders should strive to provide clarity to their team on how to manage their daily decisions on productivity and security tradeoffs as this might be new to many on their teams.

- **Between application and infrastructure teams (and cloud providers):** This relationship is undergoing significant changes because of multiple trends in the IT and security industry aimed at increasing innovation speed and developer productivity.

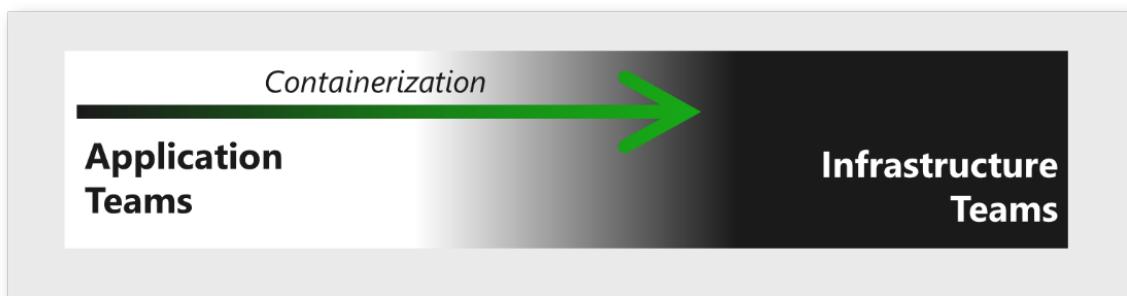
The old norms and organizational functions have been disrupted, but new norms and functions are still emerging, so we recommend accepting the ambiguity, keeping up with current thinking, and experiment with what works for your organizations until it does. We don't recommend adopting a wait-and-see approach in this space because it might put your organization at a major competitive disadvantage.

These trends are challenging the traditional norms for roles and relationships of applications and infrastructure:

- **DevOps-fusing disciplines:** In its ideal state, this effectively creates a single highly functional team that combines both sets of subject matter expertise together to rapidly innovate, release updates, and resolve issues (security and otherwise). While this ideal state will take some time to achieve and the responsibilities in the middle are still ambiguous, organizations are already reaping some benefits of rapid releases because of this cooperative approach. Microsoft recommends integrating security into this cycle to help learn those cultures, share security learnings, and work toward a common goal of rapidly releasing secure and reliable applications.



- **Containerization becoming a common infrastructure component:** Applications are increasingly being hosted and orchestrated by technologies like Docker, Kubernetes, and similar technologies. These technologies simplify development and release by abstracting many elements of the setup and configuration of the underlying operating system.



While containers began as an application development technology managed by development teams, it's becoming a common infrastructure component that is increasingly shifting to infrastructure teams. This transition is still in progress at many organizations, but it's a natural and positive direction many of the current challenges will be best solved with traditional infrastructure skill sets like networking, storage, and capacity management.

Infrastructure teams and security team members that support them should be provided with training, processes, and tooling to help manage, monitor, and secure this technology.

- **Serverless and cloud application services:** One of the dominant trends in industry right now is reducing the amount of time and development work required to build or update applications.



Developers are also increasingly using cloud services to:

- Run code instead of hosting applications on virtual machines (VMs) and servers.
- Provide application functions instead of developing their own components. This has led to a *serverless* model that uses existing cloud services for common functions. The number and variety of cloud services (and their pace of innovation) has also exceeded the ability of security teams to evaluate and approve the use of those services, leaving them to choose between allowing developers to use any service, attempting to prevent the development teams from using unapproved services, or trying to find a better way.
- **Codeless applications and Power Apps:** Another emerging trend is the use of codeless technologies like Microsoft Power Apps. This technology enables people without coding skills to create applications that achieve business outcomes. Because of this low friction and high value potential, this trend has the potential to rise in popularity quickly and security professionals would be wise to rapidly understand its implications. Security efforts should be focused on the areas where a human could make a mistake in the application, namely the design of the application and asset permissions via threat modeling the application components, interactions/relationships, and role permissions.
- **Between developers and open-source component authors:** Developers are also increasing efficiency by using open-source components and libraries instead of developing their own components. This brings value through efficiency, but also introduces security risks by creating an external dependency and a requirement to properly maintain and patch those components. Developers are effectively

assuming the risk of security and other bugs when they use these components and have to ensure there is a plan to mitigate them at the same standards as code they would develop.

- **Between applications and data:** The line between security of data and applications is becoming blurred in places and new regulations are creating a need for closer cooperation between data/privacy teams and security teams:
 - **Machine learning algorithms:** Machine learning algorithms are similar to applications in that they are designed to process data to create an outcome. The key differences are:
 - **High-value machine learning:** Machine learning often confers a significant competitive advantage and is often considered sensitive intellectual property and a trade secret.
 - **Sensitivity imprint:** Supervised machine learning is tuned using data sets, which imprints characteristics of the dataset on the algorithm. Because of this, the tuned algorithm might be considered sensitive because of the dataset used to train it. For example, training a machine learning algorithm to find secret army bases on a map using a dataset of secret army bases would make it a sensitive asset.

① Note

Not all examples are obvious, so it's critical to bring a team together with the right stakeholders from data science teams, business stakeholders, security teams, privacy teams, and others. These teams should have a responsibility to meet common goals of innovation and responsibility. They should address common issues such as how and where to store copies of data in insecure configurations, how to classify algorithms, as well as any concerns of your organizations.

Microsoft has published our [principles of responsible AI](#) to guide our own teams and our customers.

- **Data ownership and privacy:** Regulations like GDPR have increased the visibility of data issues and applications. Application teams now have the ability to control, protect, and track sensitive data at a level comparable to tracking financial data by banks and financial institutions. Data owners and applications teams need to build a rich understanding of what data applications store and what controls are required.

- **Between organizations and cloud providers:** As organizations host workloads in the cloud, they are entering into a business relationship with each of those cloud providers. The use of cloud services often brings business value such as:
 - **Accelerating digital transformation initiatives** by reducing time to market for new capabilities.
 - **Increasing value of IT and security activities** by freeing teams to focus on higher value (business-aligned) activities rather than lower-level commodity tasks that are provided more efficiently by cloud services on their behalf.
 - **Increased reliability and responsiveness:** Most modern clouds also have high uptime compared to traditional on-premises datacenters and have shown they can scale rapidly (such as during the COVID-19 pandemic) and provide resiliency following natural events like lightning strikes (which would have kept many on-premises equivalents down for much longer).

While beneficial, this shift to the cloud is not without risk. As organizations adopt cloud services, they should consider potential risk areas including:

- **Business continuity and disaster recovery:** Is the cloud provider financially healthy with a business model that's likely to survive and thrive during your organization's use of the service? Has the cloud provider made provisions to allow customer continuity if the provider experiences financial or other failure, such as providing their source code to customers or open-sourcing it?

For more information and documents regarding Microsoft's financial health, see [Microsoft investor relations](#).

- **Security:** Does the cloud provider follow industry best practices for security? Has this been validated by independent regulatory bodies?
 - [Microsoft Defender for Cloud Apps](#) allows you to discover usage of over 16,000 cloud applications, which are ranked and scored based on more than 70 risk factors to provide you with ongoing visibility into cloud use, shadow IT, and the risk that shadow IT poses to your organization.
 - The [Microsoft Service Trust Portal](#) makes regulatory compliance certifications, audit reports, pen tests, and more available to customers. These documents include many details of internal security practices (notably the SOC 2 Type 2 report and FedRAMP Moderate system security plan). [Microsoft Defender for Cloud](#) allows the [management of security policies](#) and can indicate the level of compliance with predefined industry and regulatory standards.

- **Business competitor:** Is the cloud provider a significant business competitor in your industry? Do you have sufficient protections in the cloud services contract or other means to protect your business against potentially hostile actions?

Review [this article](#) for commentary on how Microsoft avoids competing with cloud customers.

- **Multicloud:** Many organizations have a de facto or intentional multicloud strategy. This could be an intentional objective to reduce reliance on a single supplier or to access unique best of breed capabilities, but can also happen because developers chose preferred or familiar cloud services, or your organization acquired another business. Regardless of the reason, this strategy can introduce potential risks and costs that have to be managed including:
 - **Downtime from multiple dependencies:** Systems architected to rely on multiple clouds are exposed to more sources of downtime risk as disruptions in the cloud providers (or your team's use of them) could cause an outage/disruption of your business. This increased system complexity would also increase the likelihood of disruption events as team members are less likely to fully understand a more complex system.
 - **Negotiating power:** Larger organizations also should consider whether a single-cloud (mutual commitment/partnership) or multicloud strategy (ability to shift business) will achieve greater influence over their cloud providers to get their organization's feature requests prioritized.
 - **Increased maintenance overhead:** IT and security resources already are overburdened from their existing workloads and keeping up with the changes of a single cloud platform. Each additional platform further increases this overhead and takes team members away from higher value activities like streamlining technical process to speed business innovation, consulting with business groups on more effective use of technologies, and so on.
 - **Staffing and training:** Organizations often do not consider the staffing requirements necessary to support multiple platforms and the training required to maintain knowledge and currency of new features which are released in a rapid pace.

Adopt responsible and trusted AI principles

Article • 09/25/2024

The six key principles for responsible AI at Microsoft include fairness, reliability and safety, privacy and security, inclusiveness, transparency, and accountability. Use these principles to create responsible and trustworthy AI as you integrate it into mainstream products and services throughout your AI adoption journey.

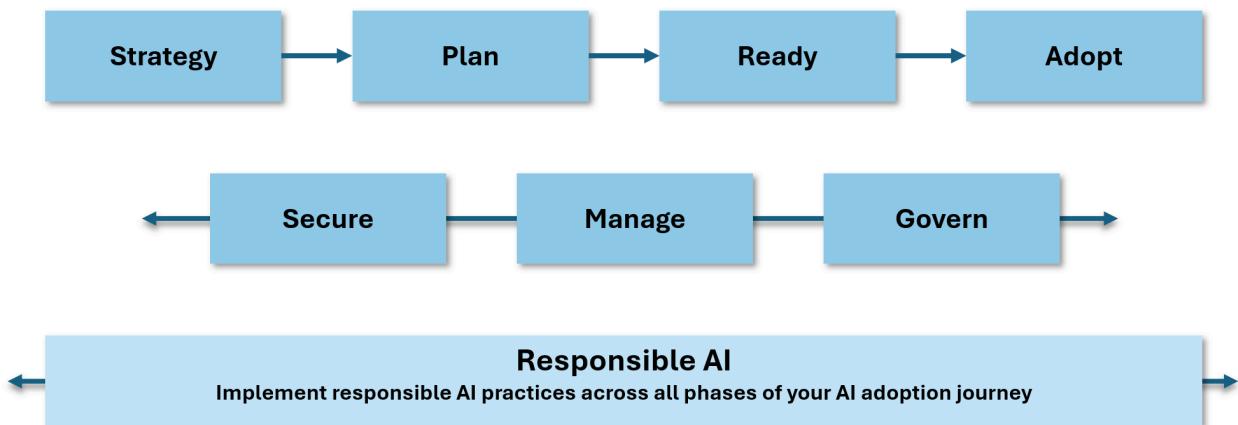
Responsible AI principles

Microsoft commits to [empowering responsible AI practices](#).

Six key principles define responsible AI:

- **Fairness:** AI systems should treat everyone equally and provide the same recommendations to all individuals. Fairness in AI systems prevents discrimination based on personal characteristics.
- **Reliability and safety:** AI systems must operate reliably, safely, and consistently under various conditions to help build trust.
- **Privacy and security:** AI systems should respect privacy and maintain security by protecting private and confidential information. They should also resist attacks and attempts to corrupt or compromise the system.
- **Inclusiveness:** AI systems should empower and engage everyone. Inclusive design practices can help AI system developers understand and address potential exclusion barriers in a product or service. Inclusiveness fosters innovation and helps design experiences that benefit everyone.
- **Transparency:** AI systems should be transparent and understandable. AI systems can inform decisions that can deeply affect people's lives, so it's crucial for individuals to understand how the system makes these decisions.
- **Accountability:** AI systems and their developers should be accountable and answerable.

Incorporate responsible AI principles throughout your AI adoption journey, from strategy and planning to implementation. Ensure that you apply these principles when you secure, manage, and govern your AI initiatives.



This AI adoption guidance includes examples of how you can use the Cloud Adoption Framework for Azure to implement responsible AI practices. The principles of responsible AI are integrated into the guidance and recommendations throughout this AI adoption journey.

The importance of responsible AI

Responsible AI helps to mitigate the following risks:

- **Unintended consequences:** Plan and oversee your responsible AI implementation to reduce the risk of unforeseen effects that have ethical implications.
- **Evolving threats:** Novel threats emerge regularly as AI technology evolves. To help mitigate and stay ahead of these threats, adhere to the principles of responsible AI.
- **Bias:** Bias mitigation in AI can be challenging but is necessary to ensure that AI systems are fair and unbiased. Use the responsible AI principles to help guide you.
- **Sensitive technologies:** Technologies like facial recognition can be considered sensitive technology because of the risk to fundamental freedoms and human rights. Consider the implications of these technologies to ensure that you use them responsibly.

Azure facilitation

Microsoft Azure provides a range of tools, services, and resources to help you build responsible AI systems.

Use Microsoft Azure AI Content Safety to build safe systems

Use [Microsoft Azure AI Content Safety](#) to detect harmful user-generated and AI-generated content in applications and services. Content Safety helps you analyze generated content in your AI applications, including text and images, to ensure that it's safe and appropriate for your users. Content Safety provides the following capabilities:

- [Prompt shields](#) scans text and documents for the risk of a user input attack, or jailbreak, on a large language model (LLM).
- [Groundedness detection](#) detects if the text responses of an LLM are grounded in the source materials that the users provide.
- [Protected material detection](#) detects if the text responses of an LLM contain protected material, such as copyrighted text, song lyrics, articles, and web content.
- The [Custom Categories \(rapid\) API](#) defines emerging harmful content patterns and scans text and images for matches.
- The [Analyze Text API](#) analyzes potentially harmful text content. It typically identifies categories like hate, self harm, and sexual or violent content.
- The [Analyze Image API](#) analyzes potential harmful image content. It typically identifies categories like hate, self harm, and sexual or violent content.

Use AI responsibly in Azure AI services

Microsoft provides a list of transparency notes for AI-relevant Azure services. The list includes services within the Azure AI services suite. For more information, see [Responsible use of AI with Azure AI services](#).

Use the Responsible AI dashboard for Azure Machine Learning

If you build systems with Azure Machine Learning, you can use the [Responsible AI dashboard](#) to assess your AI systems. The Responsible AI dashboard provides a single interface to help you implement responsible AI principles. Some of the Responsible AI features include:

- **Data analysis:** Understand and explore your dataset distributions and statistics.
- **Model overview and fairness assessment:** Evaluate your model's performance and your model's group fairness problems.
- **Error analysis:** View and understand how errors are distributed in your dataset.

- **Model interpretability:** Understand your model's predictions and how your model makes individual and overall predictions.
- **Counterfactual what-if analysis:** Observe how changes in features can affect your model predictions.
- **Causal analysis:** Use historical data to view the causal effects of treatment features on real-world outcomes.

Develop AI responsibly

 Expand table

Resource	Description
Hands-on tools for building effective human-AI experiences (HAXs) ↗	Use the HAX Toolkit early in your design process to help you conceptualize what the AI system does and how it behaves. Use the HAX Toolkit for user-facing AI products.
Conversational AI guidelines ↗	Design bots in a way that earns the trust of others to help people and society realize their full potential. Use these guidelines to create a bot that builds trust in the company and service that it represents.
Inclusive AI design guidelines ↗	Use these guidelines to help you design AI that is inclusive and accessible to everyone.
AI fairness checklist ↗	Use the AI fairness checklist to determine whether your AI system is fair and unbiased.
Responsible AI in Machine Learning	Review these responsible AI resources if you use Machine Learning to build AI systems.

Next step

Skills that you need to support the strategy phase of cloud adoption

Feedback

Was this page helpful?

 Yes

 No

How GitHub accelerates cloud adoption

Article • 05/14/2024

Overview

Innovation is the new currency in today's competitive landscape. Ride sharing, streaming content, self-driving cars, and other services have fundamentally changed people's daily rhythms while turning markets upside down and showing how the competitive landscape has moved from physical assets to digital experiences.

These types of superior digital experience are leading a disruption where well-established businesses face stiff competition from companies that can innovate and deliver value to their customers faster. To compete and avoid disruption, businesses need to build a culture of innovation and use the best and most fitting tools and cloud services.

GitHub provides a range of features that can help companies to:

- Take advantage of Azure services and capabilities.
- Modernize their practices.
- Become more agile and innovative during this cultural shift.

Companies can take advantage of GitHub's connectedness to the open-source community and find thousands of reiterated, enhanced, and ready-to-deploy cloud solution examples from organizations that have successfully adopted Azure services. They can easily borrow from and iterate on these solutions to tailor them to their business needs.

GitHub makes it easy for organizations to share within their teams, which makes it faster to modernize and deploy the next application or workload. Companies can look to *InnerSource*, a key tenet of innovation, to borrow best practices like sharing and reuse, collaboration and communication, and more from the open-source community and apply them within their organization.

From securing the open-source packages to the intellectual property that's written daily, securing the entire software supply chain should be a main priority for every company. This goal requires advanced security technology that can be incorporated and automated throughout the entire lifecycle, and native GitHub capabilities like GitHub advanced security and GitHub Actions offer this type flexibility.

Take advantage of open-source assets

Highly effective organizations recognize open-source software (OSS) as essential versus optional for modern software development. They engage with the developer communities on which they depend and use a secure platform to strategically invest in OSS. As a result, these organizations experience innovate quickly, outpace competitors, and cut costs while minimizing risk.

OSS consists of packages, libraries, scripts, and dependencies that are incorporated into applications. OSS also includes thousands of open-source assets in the form of infrastructure as code (IaC), documentation, and guidance for well-defined Azure architectures. Microsoft, partners, vendors, customers, and individuals contribute these packages to the OSS community. You can find them in GitHub, and modify, reuse, and deploy them to a specific Azure environment.

Infrastructure as code

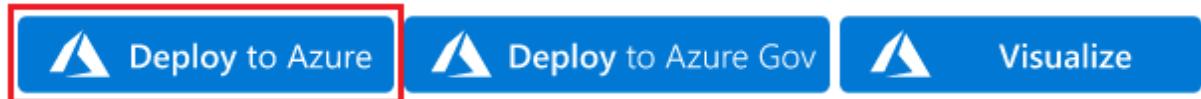
IaC is the management of infrastructure that includes networks, virtual machines, load balancers, and connection topology in a descriptive model. IaC uses the same versioning system that a DevOps team uses for source code. For example, a DevOps team follows the principle that the same source code generates the same binary. An IaC model also follows that principle and generates the same environment each time that you apply the model. IaC is a key DevOps practice that you can use with [continuous delivery \(CD\)](#).

IaC evolved to solve the problem of environment drift in the release pipeline. Without it, teams must maintain the settings of individual deployment environments, and inconsistencies between environments lead to issues during deployments. Every environment eventually becomes a snowflake, a unique configuration that can't be reproduced automatically. With snowflakes, infrastructure administration and maintenance requires manual processes that contribute to errors and are hard to track. Infrastructure deployments with IaC are repeatable and prevent runtime issues caused by configuration drift or missing dependencies.

With IaC, teams make changes to the environment description and version the configuration model, which is typically in well-documented code formats like JSON; see [Azure Resource Manager templates](#) for more information. Developers can simplify their workflows by hosting IaC code in the same GitHub repo as their application source code and adopt the same continuous integration (CI) /CD practices for IaC powered by [GitHub Actions ↗](#).

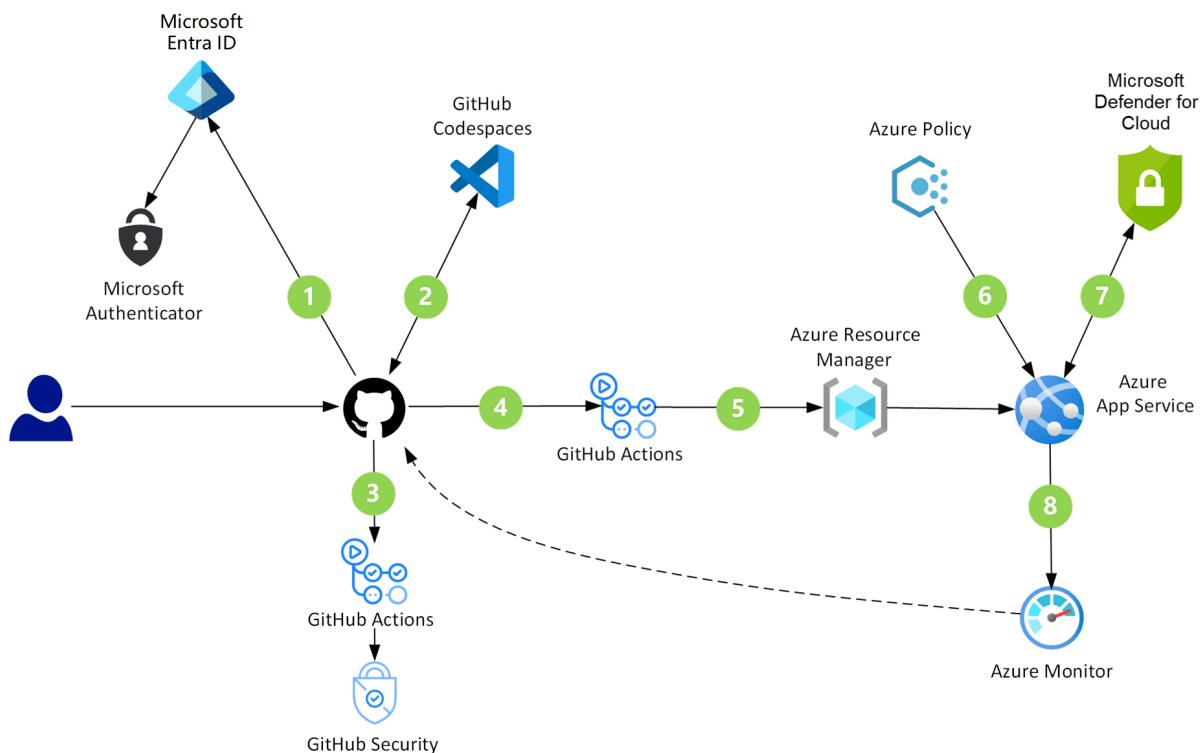
See the [AzOps ↗](#) GitHub action for an example of how to deploy custom Resource Manager templates at various Azure scopes. If you're new to Resource Manager

templates or IaC, you can also browse the [azure-quickstart-templates repo](#) on GitHub, find the template that you wish to deploy, and select the **Deploy to Azure** button to test how it works.



Cloud pattern components and best practices

The following architecture diagram highlights the security checks that run in the GitHub and Azure components of a GitHub DevSecOps environment:



- [GitHub](#) provides a code-hosting platform that developers can use for collaborating on open-source and InnerSource projects.
- [Codespaces](#) is an online development environment. Hosted by GitHub and powered by Microsoft Visual Studio Code, this tool provides a complete development solution in the cloud.
- [GitHub security](#) works to eliminate threats in multiple ways. Agents and services identify vulnerabilities in repositories and dependent packages. They also upgrade dependencies to current and secure versions.
- [GitHub Actions](#) are custom workflows that provide CI/CD capabilities directly in repositories. Computers named runners host these CI/CD jobs.

- [Microsoft Entra ID](#) is a multitenant, cloud-based identity service that controls access to Azure and other cloud applications like Microsoft 365 and GitHub.
- [Azure App Service](#) provides a framework for building, deploying, and scaling web apps. This platform offers built-in infrastructure maintenance, security patching, and scaling.
- [Azure Policy](#) helps teams manage and prevent IT issues through policy definitions that can enforce rules for cloud resources. For instance, if a project is about to deploy a virtual machine with an unrecognized SKU, Azure Policy sends alerts about the problem and stops the deployment.
- [Microsoft Defender for Cloud](#) provides unified security management and advanced threat protection across hybrid cloud workloads.
- [Azure Monitor](#) collects and analyzes performance metrics, activity logs, and other application telemetry. This service alerts applications and personnel when it identifies irregular conditions.

InnerSource

InnerSource overview

Many companies use the term *InnerSource* to describe how their engineering teams work together on code. InnerSource is a development methodology where engineers build proprietary software with best practices from large-scale open-source projects like Kubernetes or Visual Studio Code.

Large-scale open-source projects require coordination and teamwork across thousands of contributors. The most successful projects are driven by a vision for their future and daily user needs: speed, reliability, and functionality. The scale at which these projects operate provides some lessons and can help companies build better software more quickly with InnerSource.

With GitHub's pull requests and issues, collaboration and code review are built into the development process. Internal and outsourced teams can share work, discuss changes, and get feedback all in one place. This helps organizations share expertise internally and avoid reinventing field-tested solutions developed for other projects.

The anatomy of an InnerSource project

The right mix of individuals, teams, and resources can ensure a project's success. Many open-source projects follow a similar organizational structure that can help organizations to set up cross-functional teams to manage InnerSource projects. A typical open-source project has the following types of people:

- **Maintainers:** These contributors are responsible for driving the vision and managing the organizational aspects of the project. They might not be the original owners or authors of the code.
- **Contributors:** These people are everyone who has contributed something to the project.
- **Community members:** These are people who use the project. They might be active in conversations or express their opinion on the project's direction.

Bigger projects could also have subcommittees or working groups focused on different tasks like tooling, triage, and community moderation. InnerSource projects are likely to follow a similar structure. Many engineering organizations sort developers into teams like application engineering, platform engineering, and web development. Structuring organizations this way can leave blind spots that exclude qualified people. Organizing a core decision-making group supported by teams across an organization can help to rally the expertise necessary to solve problems faster.

Within an enterprise, contributors are developers across the company, and maintainers are a project's leaders and key decision-makers.

- **Maintainers:** Developers, product managers, and other key decision-makers within a company responsible for driving a project's vision and for managing day-to-day contributions.
- **Contributors:** Developers, data scientists, product managers, marketers, and other roles within a company that help drive software forward. Contributors might not be part of the direct project team but help build software by contributing code, submitting bug fixes, and more.

For more information, see the white paper [An introduction to InnerSource](#).

Automation

GitHub Actions allows users to create custom workflows directly in their GitHub repositories. Users can discover, create, and share actions to perform any job, including CI/CD, and combine actions in a completely customized workflow. They can also create

CI workflows that build and test projects written in different programming languages. Examples are available in the [guides for GitHub Actions](#).

GitHub Actions can be used to combine IaC concepts and CI/CD practices to automate the entire end-to-end deployment lifecycle, including provisioning or updating the target environment in a repeatable manner and packaging and deploying the application itself.

Example:

[GitHub Actions for Azure](#) are built to simplify how you automate your deployment processes to target Azure services such as Azure App Service, Azure Kubernetes Service, Azure Functions, and more. The [Azure starter action workflows repository](#) includes end-to-end workflows to build and deploy web apps of any language and any ecosystem to Azure. Visit [GitHub marketplace](#) to see all of the actions that are available.

Security

GitHub's shift-left security features

Starting with the first steps of development, DevSecOps adheres to security best practices. By using a shift-left strategy, DevSecOps redirects the security focus. Instead of pointing toward auditing at the end, it shifts to development in the beginning. Besides producing robust code, this fail-fast approach helps to resolve problems early when they're easy to fix.

With many security capabilities, GitHub offers tools that support every part of a DevSecOps workflow:

- Browser-based IDEs with built-in security extensions
- Agents that continuously monitor security advisories and replace vulnerable and out-of-date dependencies
- Search capabilities that scan source code for vulnerabilities
- Action-based workflows that automate every step of development, testing, and deployment
- Spaces that provide a way to privately discuss and resolve security threats and then publish the information
- Combined with the monitoring and evaluation power of Azure, these features provide a superb service for building secure cloud solutions

Example:

GitHub DevSecOps installations cover many security scenarios. Possibilities include the following cases:

- Developers who want to take advantage of preconfigured environments that offer security capabilities.
- Administrators who rely on having up-to-date, prioritized security reports at their fingertips, along with details on affected code and suggested fixes.
- Streamlined organizations that need systems to automatically acquire new and uncompromised security devices when secrets are left exposed in code.
- Development teams that could benefit from automatic upgrades when newer or more secure versions of external packages become available.

For more information, see:

- [DevSecOps in GitHub: Azure solution ideas](#)
- [Code scanning a GitHub repository using GitHub advanced security within an Azure DevOps pipeline ↗](#)
- [Applying DevSecOps to your software supply chain ↗](#)

Next steps

- Choose your implementation team (typically a developer manager and a few developers defined as admins), and deploy GitHub.
- Learn common and advanced Git workflows to enhance how you use GitHub.

The following links provide more information about GitHub.

- [Learn modules for GitHub](#)
- [GitHub Learning Lab ↗](#)
- [GitHub Docs ↗](#)

Feedback

Was this page helpful?



Skills needed to support the strategy phase of cloud adoption

Article • 02/28/2023

The objective during the strategy phase of a cloud adoption journey is to develop the plans needed to guide its technical implementation. This phase requires critical skills such as:

- Establishing your vision.
- Building your business justification.
- Rationalizing your digital estate.
- Creating your migration backlog (technical plan).

The learning paths in the sections below help you to develop each of these skills.

Establish your vision

The success of any cloud adoption effort is defined by your business vision. When the technical team doesn't understand the motives and desired outcomes, it's hard for them to guide their efforts toward business success. Read more about documenting and articulating the business vision for the technical team:

- [Adoption motivations](#). Document and articulate the reasons behind the technical effort.
- [Business outcomes](#). Clearly articulate what's expected of the technical team in terms of business changes.
- [Learning metrics](#). Establish short-term metrics that can show progress toward longer-term business outcomes.

Build your business justification

Justifying the investment to adopt the cloud can require deeper analysis and an understanding of your organization's accounting practices. The articles on business justification can help you develop these skills:

- [Cloud migration business case](#). Establish a business case for cloud migration.

Rationalize your digital estate

You can refine your business case by aligning the desired business case with current and future digital estate inventory. These articles can guide the development of a digital estate rationalization:

- [Incremental rationalization](#): An agile approach to rationalization that properly aligns late-bound technical decisions.
- [The five Rs of rationalization](#): Understand the various rationalization options.

Create your migration backlog (technical plan)

Convert your business case and rationalized digital estate into an actionable migration plan to guide the technical activities required to achieve your desired business outcomes.

Business planning skills

During the readiness phase, technical staff creates a migration landing zone capable of hosting, operating, and governing workloads that have been migrated to the cloud.

These learning paths can help you develop the necessary skills:

- [Create an Azure account](#). The first step to using Azure is to create an account. Your account holds the Azure services you provision and handles your personal settings, like identity, billing, and preferences.
- [Azure portal](#). Tour the Azure portal features and services, and customize the portal.
- [Introduction to Azure](#). Get started with Azure by creating and configuring your first virtual machine in the cloud.
- [Introduction to security in Azure](#). Learn the basic concepts for protecting your infrastructure and data when you work in the cloud. Understand what responsibilities are yours and what Azure takes care of for you.
- [Manage resources in Azure](#). Learn how to work with the Azure command line and web portal to create, manage, and control cloud-based resources.
- [Create a VM](#). Create a virtual machine by using the Azure portal.
- [Azure networking](#). Learn the basics of Azure networking and how Azure networking helps you improve resiliency and reduce latency.
- [Azure compute options](#). Learn about the Azure compute services.
- [Secure resources with Azure RBAC](#). Use Azure RBAC to secure resources.
- [Data storage options](#). Learn about the benefits of Azure data storage.

Organizational skills

Depending on the motivations and desired business outcomes of a cloud adoption effort, leaders might need to establish new organizational structures or virtual teams to facilitate various functions. These articles will help you develop the skills necessary to structure those teams to meet desired outcomes:

- [Initial organizational alignment](#). Overview of organizational alignment and various team structures to facilitate specific goals.
- [Breaking down silos and fiefdoms](#). Understanding two common organizational antipatterns and ways to guide a team to productive collaboration.

Deeper skills exploration

Various learning options beyond these initial options are available for developing skills.

Typical mappings of cloud IT roles

Microsoft and partners offer various options for all audiences to develop skills with Azure services.

- [Map roles and skills](#): A resource for mapping your cloud career path. Learn about your cloud role and suggested skills. Follow a learning curriculum at your own pace to build the skills that you need most to stay relevant.
- Explore [Azure certification training and exams](#) to gain official recognition for your Azure knowledge.

Microsoft Learn

Microsoft Learn is a new approach to learning. Readiness for the new skills and responsibilities that come with cloud adoption doesn't come easily. Microsoft Learn provides a more rewarding approach to hands-on learning that helps you achieve your goals faster. Earn points and levels, and achieve more!

Here is an example of a tailored learning path that aligns to the Strategy methodology of the Cloud Adoption Framework.

[Learn the business value of Microsoft Azure](#): This learning experience will take you on a journey that will begin by showing you how digital transformation and the power of the cloud can transform your business. We will cover how Microsoft Azure Cloud Services can power your organization on a trusted cloud platform. Finally, we will wrap up by illustrating how to make this journey real for your organization.

Learn more

To discover more learning paths, browse the [Microsoft Learn training catalog](#). Use the **Roles** filter to align learning paths with your role.

Cloud strategy antipatterns

Article • 03/22/2023

Customers often experience antipatterns during the Strategy phase of cloud adoption. These antipatterns can complicate the alignment of IT and business strategies. These antipatterns also make measuring the success of cloud projects more difficult.

Antipattern: Adopt the cloud without establishing goals

Many companies announce cloud-first or cloud-only strategies. But, they don't clearly define what they want to achieve with those strategies. Few cloud projects can succeed without concrete KPIs and goals. It's impossible to measure project performance without indicators or specified targets.

Example: Migrate to the cloud without defining goals

A corporation's closest competitor launches a cloud-only strategy. The competitor's goal is to accelerate business by having all systems in the cloud within a year. The corporation doesn't want to trail behind. The corporation's directors begin strategic discussions on how to adopt the cloud quickly. But, they don't define any concrete success criteria like reducing costs or improving system performance.

The corporation's first system migrates to the cloud. The directors can't check whether their cloud strategy is successful because they never defined what they wanted to achieve.

Preferred outcome: Define goals and KPIs

Define concrete KPIs when discussing your reasons for adopting the cloud. Then you'll be able to measure how successful your strategy is. You'll also know whether you can use the same strategy for other projects. See [Why are we moving to the cloud?](#) to learn more about motivations for cloud adoption.

Antipattern: Fail to communicate motivations

Cloud adoption journeys can fail when motivations or cloud adoption triggers are misaligned within a company. A business might see major benefits in adopting the cloud but fail to communicate these adoption triggers to IT. This problem even comes

up when these motivations influence the company's IT strategy, not just its business strategy. Without alignment or documented motivations, cloud journeys often fail.

Example: Fail to communicate benefits

A corporation starts using the cloud when its managing board announces a cloud-first IT strategy. But, the board doesn't explain how the strategy benefits the company. The IT and business departments aren't certain why they're adopting the cloud. This uncertainty leads to a lack of focus, meaning the departments fail to work toward this common goal. Hesitancy about adopting the cloud increases, especially within IT. Because the IT strategy has poorly defined goals, that strategy comes under scrutiny, leading to more questions than answers.

Preferred outcome: Define and communicate reasons for cloud adoption

Decide why you want to adopt the cloud. Then clearly define your reasons and communicate them throughout the company. Your IT and business departments will then accept cloud adoption more readily. You can also use these motivations to influence technical decisions in later stages of the cloud adoption journey. Before justifying a move to the cloud, review [cloud migration myths](#).

Next steps

- [Why are we moving to the cloud?](#)
- [Build a business justification for cloud migration](#)

Develop a cloud adoption plan

Article • 02/28/2023

Cloud adoption plans convert the aspirational goals of a cloud adoption strategy into an actionable plan. Your collective cloud teams can use the cloud adoption plan to guide their technical efforts and align them with your organization's business strategy.

Use the following exercises to help you document your organization's technology strategy. These exercises support cloud adoption efforts by capturing prioritized tasks. At the end of this process, your cloud adoption plan will map to the metrics and motivations defined in the cloud adoption strategy.

Exercises	
1	Digital estate : Inventory and rationalize your digital estate based on assumptions that align your organization's motivations and business outcomes.
2	Initial organizational alignment : Establish a plan for initial organizational alignment to support the adoption plan.
3	Skills readiness plan : Create a plan for addressing skills readiness gaps within your organization.
4	Cloud adoption plan : Develop a cloud adoption plan to manage change across skills, the digital estate, and your organization.

Download the [strategy and plan template](#) to track the outputs of each exercise as you build out your cloud adoption strategy. Also, learn about the [five Rs of cloud rationalization](#) to help build your cloud adoption plan.

Cloud rationalization

Article • 06/07/2023

Cloud rationalization is the process of evaluating assets to determine the best way to migrate or modernize each asset in the cloud. For more information about the process of rationalization, see [What is a digital estate](#).

Rationalization context

The *five Rs of rationalization* listed in this article are a great way to label a potential future state for any workload that's being considered as a cloud candidate. Put this labeling process into the correct context before you attempt to rationalize an environment. To provide that context, review the following myths:

Myth: It's easy to make rationalization decisions early in the process

Good rationalization requires a deep knowledge of the workload and associated assets, like applications, infrastructure, and data. Most importantly, good rationalization decisions take time. We recommend that you use an [incremental rationalization process](#).

Myth: Cloud adoption has to wait for all workloads to be rationalized

When an entire IT portfolio or even a single datacenter is rationalized, it can delay the realization of business value by months or even years. Avoid full rationalization when possible. Instead, use the [Power of 10 approach to release planning](#) to make wise decisions about the next 10 workloads that are slated for cloud adoption.

Myth: Business justification has to wait for all workloads to be rationalized

To develop a business justification for a cloud adoption effort, make a few basic assumptions at the portfolio level. When motivations are aligned to innovation, assume rearchitecture. If they're aligned to migration, assume rehost. These assumptions can accelerate the business justification process. During the assessment phase of each workload's adoption cycle, assumptions are then challenged, and budgets are refined.

Now review the following five Rs of rationalization to familiarize yourself with the long-term process. While developing your cloud adoption plan, choose the option that best aligns with your motivations, business outcomes, and current state environment. The goal in digital estate rationalization is to set a baseline, not to rationalize every workload.

The five Rs of rationalization

The following five Rs of rationalization describe the most common options for rationalization.

Rehost

Also known as a *lift and shift* migration, a rehost effort moves a current state asset to the chosen cloud provider with minimal change to the overall architecture.

Common drivers might be to:

- Reduce capital expense.
- Free up datacenter space.
- Achieve rapid return on investment in the cloud.

Quantitative analysis factors are:

- VM size, including CPU, memory, and storage.
- Dependencies, like network traffic.
- Asset compatibility.

Qualitative analysis factors are:

- Tolerance for change.
- Business priorities.
- Critical business events.
- Process dependencies.

Refactor

Platform as a service (PaaS) options can reduce the operational costs that are associated with many applications. It's a good idea to slightly refactor an application to fit a PaaS-based model.

Refactor also refers to the application development process of refactoring code to enable an application to deliver on new business opportunities.

Common drivers might include:

- Faster and shorter updates.
- Code portability.
- Greater cloud efficiency to help with resources, speed, cost, and managed operations.

Quantitative analysis factors are:

- Application asset size, like CPU, memory, and storage.
- Dependencies, like network traffic.
- User traffic, like page views, time on page, and load times.
- Development platforms, like languages, data platforms, and middle tier services.
- Database that includes CPU, memory, storage, and version.

Qualitative analysis factors are:

- Continued business investments.
- Bursting options or timelines.
- Business process dependencies.

Rearchitect

Some aging applications aren't compatible with cloud providers. This incompatibility is because of the architectural decisions that were made when the application was built. In these cases, the application might need to be rearchitected before transformation.

In other cases, applications that are cloud-compatible, but not cloud-native, might create cost efficiencies and operational efficiencies by rearchitecting the solution into a cloud-native application.

Common drivers might include:

- Application scale and agility.
- Easier adoption of new cloud capabilities.
- Mix of technology stacks.

Quantitative analysis factors are:

- Application asset size, like CPU, memory, and storage.
- Dependencies, like network traffic.

- User traffic, like page views, time on page, and load times.
- Development platforms, like languages, data platforms, and middle tier services.
- Database that includes CPU, memory, storage, and version.

Qualitative analysis factors are:

- To grow business investments.
- Operational costs.
- Potential feedback loops and DevOps investments.

Rebuild

In some scenarios, the delta that must be overcome to carry an application forward can be too large to justify further investment. This issue is especially true for applications that previously met the needs of a business but are now unsupported with the current business processes. To resolve the issue, create a new code base to align with a [cloud-native](#) approach.

Common drivers might be to:

- Accelerate innovation.
- Build applications faster.
- Reduce operational cost.

Quantitative analysis factors are:

- Application asset size, like CPU, memory, and storage.
- Dependencies, like network traffic.
- User traffic, like page views, time on page, and load times.
- Development platforms, like languages, data platforms, and middle tier services.
- Database that includes CPU, memory, storage, and version.

Qualitative analysis factors are:

- Declining end user satisfaction.
- Business processes that are limited by functionality.
- Potential cost, experience, or revenue gains.

Replace

Solutions are typically implemented by using the best technology and approach available at the time. Sometimes software as a service (SaaS) applications can provide all

the necessary functionality for the hosted application. In these scenarios, a workload can be scheduled for future replacement, which removes it from the transformation effort.

Common drivers might be to:

- Standardize around industry best practices.
- Accelerate the adoption of business process-driven approaches.
- Reallocate development investments into applications that create competitive differentiation or advantages.

Quantitative analysis factors are:

- General operating cost reductions.
- VM size, including CPU, memory, and storage.
- Dependencies, like network traffic.
- Assets to be retired.
- Database that includes CPU, memory, storage, and version.

Qualitative analysis factors are:

- Cost benefit analysis of the current architecture versus a SaaS solution.
- Business process maps.
- Data schemas.
- Custom or automated processes.

Next steps

You can apply these five Rs of rationalization to a digital estate to help you make rationalization decisions about the future state of each application.

[What is a digital estate](#)

What is a digital estate?

Article • 04/28/2023

Every modern company has some form of digital estate. Much like a physical estate, a digital estate is an abstract reference to a collection of tangible owned assets. In a digital estate, those assets include virtual machines (VMs), servers, applications, data, and so on. Essentially, a digital estate is the collection of IT assets that power business processes and supporting operations.

The importance of a digital estate is most obvious during the planning and execution of digital transformation efforts. During transformation journeys, the cloud strategy teams use the digital estate to map the business outcomes to release plans and technical efforts. That all starts with an inventory and measurement of the digital assets that the organization owns today.

How can a digital estate be measured?

The measurement of a digital estate changes depending on the desired business outcomes.

- **Infrastructure migrations:** When an organization is inward-facing and seeks to optimize costs, operational processes, agility, or other aspects of their operations, the digital estate focuses on VMs, servers, and workloads.
- **Application innovation:** For customer-focused transformations, the lens is a bit different. The focus should be placed on the applications, APIs, and transactional data that supports the customers. VMs and network appliances often receive less focus.
- **Data-driven innovation:** In today's digitally driven market, it's difficult to launch a new product or service without a strong foundation in data. During cloud-enabled data innovation efforts, the focus is more on the silos of data across the organization.
- **Operational stability:** Businesses are dependent on stable technologies to operate effectively. Near-zero downtime and service reliability are crucial in competitive markets. When operational stability is a priority, the digital estate should be measured on positive or negative impact to stable operations. Business continuity, disaster recovery, and reliability of workloads and each asset are required measures when operational stability is a priority

- **Sustainable platform:** With a global drive towards sustainability, businesses might want to move towards a more sustainable platform to reduce carbon emissions.

After an organization understands the most important form of transformation, digital estate planning becomes much easier to manage.

Each type of transformation can be measured with any of the above views. Companies commonly complete all these transformations in parallel. We strongly recommend that company leadership and the cloud strategy team agree regarding the transformation that is most important for business success. That understanding serves as the basis for common language and metrics across multiple initiatives.

How can a financial model be updated to reflect the digital estate?

An analysis of the digital estate drives cloud adoption activities. It also informs financial models by providing cloud costing models, which in turn drive return on investment (ROI).

To complete the digital estate analysis, take the following steps:

1. [Determine analysis approach](#).
2. [Collect current state inventory](#).
3. [Rationalize the assets in the digital estate](#).
4. [Align assets to cloud offerings to calculate pricing](#).

Financial models and migration backlogs can be modified to reflect the rationalized and priced estate.

Next steps

Before digital estate planning begins, determine which approach to use.

[Approaches to digital estate planning](#)

Approaches to digital estate planning

Article • 02/10/2023

Digital estate planning takes several forms depending on the desired outcomes and size of the existing estate. There are various approaches that you can take. It's important to set expectations regarding the approach early in planning cycles. Unclear expectations often lead to delays associated with other inventory-gathering exercises. This article outlines three approaches to analysis.

Workload-driven approach

The top-down assessment approach evaluates security aspects. Security includes data categorization (high, medium, or low business impact), compliance, sovereignty, and security risk requirements. This approach assesses high-level architectural complexity. It evaluates aspects such as authentication, data structure, latency requirements, dependencies, and application life expectancy.

The top-down approach measures the operational requirements of the application, such as service levels, integration, maintenance windows, monitoring, and insight. When these aspects have been analyzed and considered, the resulting score reflects the relative difficulty of migrating this application to each cloud platform: IaaS, PaaS, and SaaS.

The top-down assessment evaluates the financial benefits of the application, such as operational efficiencies, TCO, return on investment, and other appropriate financial metrics. The assessment also examines the seasonality of the application (such as whether there are certain times of the year when demand spikes) and overall compute load.

It also looks at the types of users it supports (casual or expert, always or occasionally logged on), and the required scalability and elasticity. Finally, the assessment concludes by examining business continuity and resiliency requirements, and dependencies for running the application if a disruption of service should occur.

💡 Tip

The workload-driven approach requires interviews and anecdotal feedback from business and technical stakeholders. Availability of key individuals is the biggest risk to timing. The anecdotal nature of the data sources makes it more difficult to

produce accurate cost or timing estimates. Plan schedules in advance and validate any data that's collected.

Asset-driven approach

The asset-driven approach provides a plan based on the assets that support an application for migration. In this approach, you pull statistical usage data from a configuration management database (CMDB) or other infrastructure assessment tools.

This approach usually assumes an IaaS model of deployment as a baseline. In this process, the analysis evaluates the attributes of each asset:

- Memory
- Number of processors (CPU cores)
- Operating system storage space
- Data drives
- Network interface cards (NICs)
- IPv6
- Network load balancing
- Clustering
- Operating system version
- Database version (if necessary)
- Supported domains
- Third-party components or software packages, among others

The assets that you inventory in this approach are then aligned with workloads or applications for grouping and dependency mapping purposes.

Tip

The asset-driven approach requires a rich source of statistical usage data. The time that's needed to scan the inventory and collect data is the biggest risk to timing. The low-level data sources can miss dependencies between assets or applications. Plan for at least one month to scan the inventory. Validate dependencies before deployment.

Incremental approach

We strongly suggest an incremental approach, as we do for many processes in the Cloud Adoption Framework. With digital estate planning, that equates to a multiphase

process:

- **Initial cost analysis:** If you require financial validation, start with an asset-driven approach, described earlier, to get an initial cost calculation for the entire digital estate, with no rationalization. This approach establishes a worst-case scenario benchmark.
- **Migration planning:** After you've assembled a cloud strategy team, build an initial migration backlog using a workload-driven approach that's based on their collective knowledge and limited stakeholder interviews. This approach quickly builds a lightweight workload assessment to foster collaboration.
- **Release planning:** At each release, prune and reprioritize the migration backlog to focus on the most relevant business impact. During this process, you select the next 5 to 10 workloads as prioritized releases. At this point, the cloud strategy team invests the time in completing an exhaustive workload-driven approach. Delaying this assessment until a release better respects the time of stakeholders. It also delays the investment in full analysis until the business starts to see results from earlier efforts.
- **Implementation analysis:** Before migrating, modernizing, or replicating any asset, assess it both individually and as part of a collective release. At this point, you can scrutinize the data from the initial asset-driven approach to ensure accurate sizing and operational constraints.

Tip

The incremental approach enables streamlined planning and accelerated results. It's important that all parties involved understand the approach to delayed decision making. It's equally important that assumptions made at each stage be documented to avoid loss of details.

Next steps

After you choose an approach, gather the inventory data.

Gather inventory data

Gather inventory data for a digital estate

Article • 12/01/2022

Developing an inventory is the first step for [digital estate planning](#). In this process, a list of IT assets that support specific business functions are collected for later analysis and rationalization. This article assumes that a bottom-up approach to analysis is most appropriate for planning. For more information, see [Approaches to digital estate planning](#).

Take inventory of a digital estate

The inventory that supports a digital estate changes depending on the desired digital transformation and corresponding transformation journey.

- **Cloud migration:** We often recommend that during a cloud migration, you collect the inventory from scanning tools that create a centralized list of all virtual machines and servers. Some tools can also create network mappings and dependencies, which help define workload alignment.
- **Application innovation:** Inventory during a cloud-enabled application innovation effort begins with the customer. Mapping the customer experience from start to finish is a good place to begin. Aligning that map to applications, APIs, data, and other assets creates a detailed inventory for analysis.
- **Data innovation:** Cloud-enabled data innovation efforts focus on the product or service. An inventory also includes a mapping of the opportunities for disrupting the market, as well as the capabilities needed.
- **Security:** Inventory provides security the understanding to help assess, protect, and monitor the organization's assets.

Accuracy and completeness of an inventory

An inventory is rarely complete in its first iteration. We strongly recommend the cloud strategy team aligns stakeholders and power users to validate the inventory. When possible, use additional tools like network and dependency analysis to identify assets that are being sent traffic, but that are not in the inventory.

Next steps

After an inventory is compiled and validated, it can be rationalized. Inventory rationalization is the next step to digital estate planning.

[Rationalize the digital estate](#)

Rationalize the digital estate

Article • 12/01/2022

Cloud rationalization is the process of evaluating assets to determine the best approach to hosting them in the cloud. After you've determined an [approach](#) and aggregated an [inventory](#), cloud rationalization can begin. Cloud rationalization discusses the most common rationalization options.

Watch the following video to get a quick overview about completing a comprehensive assessment that will help you plan and prioritize your migration efforts.

<https://www.microsoft.com/en-us/videoplayer/embed/RWDpel?postJs||Msg=true> ↗

Traditional view of rationalization

It's easy to understand rationalization when you visualize the traditional process of rationalization as a complex decision tree. Each asset in the digital estate is fed through a process that results in one of five answers (the five Rs of rationalization). For small estates, this process works well. For larger estates, it's inefficient and can lead to significant delays. Let's examine the process to see why. Then we'll present a more efficient model.

Inventory: A thorough inventory of assets, including applications, software, hardware, operating systems, and system performance metrics, is required for completing a full rationalization by using traditional models.

Quantitative analysis: In the decision tree, quantitative questions drive the first layer of decisions. Common questions include the following:

- Is the asset in use today?
- If so, is it optimized and sized properly?
- What dependencies exist between assets? These questions are vital to the classification of the inventory.

Qualitative analysis: The next set of decisions requires human intelligence in the form of qualitative analysis. Often, the questions that come up here are unique to the solution and can be answered only by business stakeholders and power users. These decisions typically delay the process, slowing things down considerably. This analysis generally consumes 40 to 80 FTE hours per application.

For guidance about building a list of qualitative analysis questions, see [Approaches to digital estate planning](#).

Rationalization decision: In the hands of an experienced rationalization team, the qualitative and quantitative data creates clear decisions. Unfortunately, teams with a high degree of rationalization experience are expensive to hire or take months to train.

Rationalization at enterprise-scale

If this effort is time consuming and daunting for a 50-VM digital estate, imagine the effort that's required to drive business transformation in an environment with thousands of VMs and hundreds of applications. The human effort required can easily exceed 1,500 FTE hours and nine months of planning.

While full rationalization is the end state and a great direction to move in, it seldom produces a high ROI (return on investment) relative to the time and energy that's required.

When rationalization is essential to financial decisions, it's worth considering a professional services organization that specializes in cloud rationalization to accelerate the process. Even then, full rationalization can be a costly and time-consuming effort that delays transformation or business outcomes.

The rest of this article describes an alternative approach, known as incremental rationalization.

Incremental rationalization

The complete rationalization of a large digital estate is prone to risk and can suffer delays because of its complexity. The assumption behind the incremental approach is that delayed decisions stagger the load on the business to reduce the risk of roadblocks. Over time, this approach creates an organic model for developing the processes and experience required to make qualified rationalization decisions more efficiently.

Inventory: Reduce discovery data points

Few organizations invest the time, energy, and expense in maintaining an accurate real-time inventory of the full digital estate. Loss, theft, refresh cycles, and employee onboarding often justify detailed asset tracking of end-user devices. The ROI of maintaining an accurate server and application inventory in a traditional, on-premises datacenter is often low. Most IT organizations have more urgent issues to address than tracking the usage of fixed assets in a datacenter.

In a cloud transformation, inventory directly correlates to operating costs. Accurate inventory data is required for proper planning. Unfortunately, current environmental scanning options can delay decisions by weeks or months. Fortunately, a few tricks can accelerate data collection.

Agent-based scanning is the most frequently cited delay. The robust data that's required for a traditional rationalization can often only be collected with an agent running on each asset. This dependency on agents often slows progress, because it can require feedback from security, operations, and administration functions.

In an incremental rationalization process, an agentless solution could be used for an initial discovery to accelerate early decisions. Depending on the level of complexity in the environment, an agent-based solution might still be required, but it can be removed from the critical path to business change.

Quantitative analysis: Streamline decisions

Regardless of the approach to inventory discovery, quantitative analysis can drive initial decisions and assumptions. This is especially true when trying to identify the first workload or when the goal of rationalization is a high-level cost comparison. In an incremental rationalization process, the cloud strategy team and the cloud adoption teams limit the [five Rs of rationalization](#) to two concise decisions and only apply those quantitative factors. This streamlines the analysis and reduces the amount of initial data that's required to drive change.

For example, if an organization is in the midst of an IaaS migration to the cloud, you can assume that most workloads will either be retired or rehosted.

Qualitative analysis: Temporary assumptions

By reducing the number of potential outcomes, it's easier to reach an initial decision about the future state of an asset. When you reduce the options, you also reduce the number of questions asked of the business at this early stage.

For example, if the options are limited to rehosting or retiring, the business needs to answer only one question during initial rationalization, which is whether to retire the asset.

"Analysis suggests that no users are actively using this asset. Is that accurate, or have we overlooked something?" Such a binary question is typically much easier to run through qualitative analysis.

This streamlined approach produces baselines, financial plans, strategy, and direction. In later activities, each asset goes through further rationalization and qualitative analysis to evaluate other options. All assumptions that you make in this initial rationalization are tested before migrating individual workloads.

Challenge assumptions

The outcome of the prior section is a rough rationalization that's full of assumptions. Next, it's time to challenge some of those assumptions.

Retire assets

In a traditional on-premises environment, hosting small, unused assets seldom causes a significant impact on annual costs. With a few exceptions, FTE effort that's required to analyze and retire the actual asset outweighs the cost savings from pruning and retiring those assets.

When you move to a cloud accounting model, retiring assets can produce significant savings in annual operating costs and up-front migration efforts.

It's not uncommon for organizations to retire 20% or more of their digital estate after completing a quantitative analysis. We recommend conducting further qualitative analysis before taking action. After it's confirmed, retiring those assets can produce the first ROI victory of the cloud migration. This is often one of the biggest cost-saving factors. Therefore, the cloud strategy team should oversee the validation and retirement of assets, in parallel with execution of the [Migrate methodology](#), to achieve an early financial win.

Program adjustments

A company seldom embarks on just one transformation journey. The choice between cost reduction, market growth, and new revenue streams is rarely a binary decision. As such, we recommend that the cloud strategy team work with IT to identify assets on parallel transformation efforts that are outside of the scope of the primary transformation journey.

In the IaaS migration example given in this article:

- Ask the DevOps team to identify assets that are already part of a deployment automation and remove those assets from the core migration plan.

- Ask the data and R&D teams to identify assets that are powering new revenue streams and remove them from the core migration plan.

This program-focused qualitative analysis can be executed quickly and creates alignment across multiple migration backlogs.

You might still need to consider some assets as rehost assets for a while. You can phase in later rationalization after the initial migration.

Select the first workload

Implementing the first workload is key to testing and learning. It's the first opportunity to demonstrate and build a growth mindset.

Business criteria

To ensure business transparency, identify a workload that is supported by a member of the cloud strategy team's business unit. Preferably choose one in which the team has a vested stake and strong motivation to move to the cloud.

Technical criteria

Select a workload that has minimum dependencies and can be moved as a small group of assets. We recommend that you select a workload with a defined testing path to make validation easier.

The first workload is often deployed in an experimental environment with no operational or governance capacity. It's important to select a workload that doesn't interact with secure data.

Qualitative analysis

The cloud adoption teams and the cloud strategy team can work together to analyze this small workload. This collaboration creates a controlled opportunity to create and test qualitative analysis criteria. The smaller population creates an opportunity to survey the affected users, and to complete a detailed qualitative analysis in a week or less. For common qualitative analysis factors, see the specific rationalization target in the [five Rs of rationalization](#).

Migration

In parallel with continued rationalization, the cloud adoption team can begin migrating the small workload to expand learning in the following key areas:

- Strengthen skills with the cloud provider's platform.
- Define the core services and Azure standards needed to fit the long-term vision.
- Better understand how operations might need to change later in the transformation.
- Understand any inherent business risks and the business's tolerance for those risks.
- Establish a baseline or minimum viable product (MVP) for governance based on the business's risk tolerance.

Release planning

While the cloud adoption team is executing the migration or implementation of the first workload, the cloud strategy team can begin prioritizing the remaining applications and workloads.

Power of 10

The traditional approach to rationalization attempts to meet all foreseeable needs. Fortunately, a plan for every application is often not required to start a transformation journey. In an incremental model, the Power of 10 approach provides a good starting point. In this model, the cloud strategy team selects the first 10 applications to be migrated. Those ten workloads should contain a mixture of simple and complex workloads.

Build the first backlogs

The cloud adoption teams and the cloud strategy team can work together on the qualitative analysis for the first 10 workloads. This effort creates the first prioritized migration backlog and the first prioritized release backlog. This method enables the teams to iterate on the approach and provides sufficient time to create an adequate process for qualitative analysis.

Mature the process

After the two teams agree on the qualitative analysis criteria, assessment can become a task within each iteration. Reaching consensus on assessment criteria usually requires two to three releases.

After the assessment has moved into the incremental execution process of migration, the cloud adoption team can iterate faster on assessment and architecture. At this stage, the cloud strategy team is also abstracted, reducing the drain on their time. This also enables the cloud strategy team to focus on prioritizing the applications that are not yet in a specific release, ensuring tight alignment with changing market conditions.

Not all of the prioritized applications will be ready for migration. Sequencing is likely to change as the team does deeper qualitative analysis and discovers business events and dependencies that might prompt reprioritization of the backlog. Some releases might group together a small number of workloads. Others might just contain a single workload.

The cloud adoption team is likely to run iterations that don't produce a complete workload migration. The smaller the workload, and the fewer dependencies, the more likely a workload is to fit into a single sprint or iteration. For this reason, we recommend that the first few applications in the release backlog be small and contain few external dependencies.

End state

Over time, the cloud adoption team and the cloud strategy team together complete a full rationalization of the inventory. This incremental approach enables the teams to get continually faster at the rationalization process. It also helps the transformation journey to yield tangible business results sooner, without as much upfront analysis effort.

In some cases, the financial model might be too tight to make a decision without additional rationalization. In such cases, you might need a more traditional approach to rationalization.

Next steps

The output of a rationalization effort is a prioritized backlog of all assets that are affected by the chosen transformation. This backlog is now ready to serve as the foundation for costing models of cloud services.

[Align cost models with the digital estate](#)

Align cost models with the digital estate to forecast cloud costs

Article • 05/22/2024

After you've rationalized a digital estate, you can align it to equivalent costing models with the chosen cloud provider. Discussing cost models is difficult without focusing on a specific cloud provider. To provide tangible examples in this article, Azure is the assumed cloud provider.

Azure pricing tools help you manage cloud spend with transparency and accuracy, so you can make the most of Azure and other clouds. Providing the tools to monitor, allocate, and optimize cloud costs, empowers customers to accelerate future investments with confidence.

- [Azure Migrate](#): Azure Migrate is perhaps the most cost effective approach to cost model alignment. This tool allows for [digital estate inventory](#), [limited rationalization](#), and cost calculations in one tool.
- [Total cost of ownership \(TCO\) calculator](#): Lower the total cost of ownership of your on-premises infrastructure with the Azure cloud platform. Use the Azure TCO calculator to estimate the cost savings you can realize by migrating your application workloads to Azure. Provide a brief description of your on-premises environment to get an instant report.
- [Azure pricing calculator](#): Estimate your expected monthly bill by using our pricing calculator. Track your actual account usage and bill at any time using the billing portal. Set up automatic email billing alerts to notify you if your spend goes above an amount you configure.
- [Microsoft Cost Management](#): Microsoft Cost Management is a cost management solution that helps you use and manage Azure and other cloud resources effectively. Collect cloud usage and billing data through application program interfaces (APIs) from Azure, Amazon Web Services, and Google Cloud Platform. With that data, gain full visibility into resource consumption and costs across cloud platforms in a single, unified view. Continuously monitor cloud consumption and cost trends. Track actual cloud spending against your budget to avoid overspending. Detect spending anomalies and usage inefficiencies. Use historical data to improve your forecasting accuracy for cloud usage and expenditures.

Feedback

Was this page helpful?

 Yes

 No

Measure business outcomes by using AppDynamics

Article • 01/09/2023

Measuring and quantifying business outcomes is a crucial part of any cloud adoption strategy. Understanding an application's performance and user experience is key to measuring those business outcomes. However, accurately measuring the correlation between application performance, user experience, and business impact is often difficult and time consuming.

AppDynamics can provide business insights for most metrics. Many organizations start a comprehensive cloud adoption strategy with these metrics:

- A pre-migration and post-migration comparison
- Business health
- Release validation
- Segment health
- User journeys
- Business journeys
- Conversion funnels

This article describes how to measure the business outcomes of a cloud adoption migration. It also describes how to speed up a migration and reduce risks.

How AppDynamics works

To use AppDynamics, you deploy a small, lightweight agent alongside your applications before your migration. Agents support various languages, like .NET, Java, and Node.js. The agent collects performance and diagnostic data during the migration and sends it to a controller that correlates and analyzes the information. Controllers can reside in a fully managed AppDynamics environment, or you can manage them in Azure. Key user experiences are identified as *business transactions*, which help you discover the baseline for normal application or business performance. Whether they're traditional server infrastructure, databases, middleware components, on-premises, or in the cloud, all application components and dependencies are identified in real time for the entire application and each business transaction.

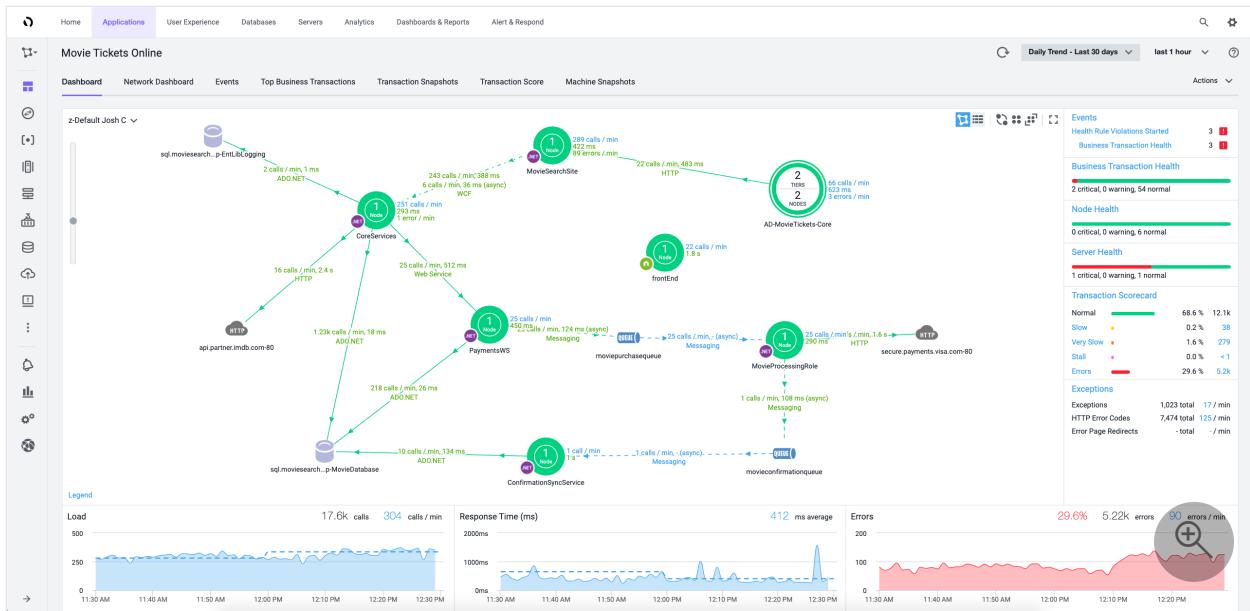


Figure 1: AppDynamics flow map.

AppDynamics and business metrics

AppDynamics helps you define business value for your applications, identify the key metrics that they should meet to retain their value, and determine whether they're meeting their target business outcomes. AppDynamics agents collect these data points and traditional application performance metrics like response time and memory utilization in real time, directly from the application, without any changes to code.

Business metrics are closely related to business outcomes. Many organizations have complex metrics that measure unique business outcomes. These outcomes can range from fiscal and agility-related metrics to performance and customer engagement goals. AppDynamics collects the metrics that are specific and useful to your organization. Those metrics can contribute to business operations before and after you migrate workloads to Azure.

Example:

A company that sells widgets in an online marketplace identifies these key business transactions in its web application:

- Landing page
- Add to cart
- Shipping
- Billing
- Confirm order

These business transactions are common to e-commerce applications. A *conversion funnel* is the journey that users take when they move through these pages. It leads directly to sales revenue for the company. When users abandon the journey because of poor page performance or errors, it affects the company's profits.

The company identifies these additional key business metrics:

- Cart totals
- Customer segments
- Customer locations

Combining application and business performance metrics helps to demonstrate how the application's performance relates to profit. These insights are vital during migrations.

Configurable dashboards are one of many AppDynamics tools that visualize these insights. In the following example, you can see the overall conversion funnel and the effect of individual page performance on abandoners. You can also see shopping cart totals, customer segments, location, and general revenue details.

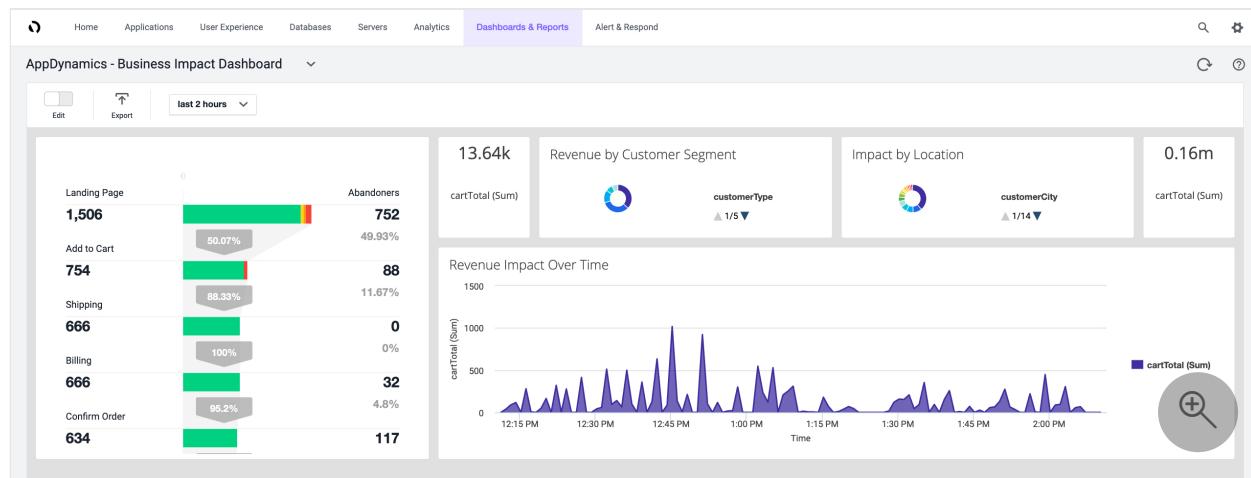


Figure 2: AppDynamics business impact dashboard.

Identify business metrics

The [strategy](#) and [business outcomes](#) sections of the Cloud Adoption Framework for Azure provide strategies to help you identify business outcomes for your organization.

Pre-migration and post-migration comparison

The cloud offers extensive benefits and potential, but the first steps of a migration are often unclear and risky. You need to use more criteria, beyond the success of the deployment, to evaluate a migration. Understanding the user experience and business performance before and after cloud migration helps you to adjust and stabilize both as

needed. Those adjustments help you produce successful business outcomes while reinforcing the value that Azure provides throughout your migration.

To build on the foundation that AppDynamics provides, compare business and application metrics before and after a migration to evaluate whether the target business outcomes are met.

Example:

Movie Tickets Online, a fictional online ticket vendor, is retiring its existing datacenters and moving its workloads to Azure. Capacity problems have led to poor business transaction performance, and the company looks forward to the performance optimizations and capacity provided by Azure.

In addition to improving performance, the company wants to ensure that the business outcome goals of improving sales funnels and growing revenue are met. As part of the migration, the company deployed AppDynamics to its existing on-premises environments to clearly understand the current performance. As part of the cloud deployment, the company can use the AppDynamics native integration with Azure to understand post-migration performance and business outcomes.

Movie Tickets Online experienced an increase in conversion rates from 48 to 79 percent and improvements to performance, response time, and ticket sales volume.

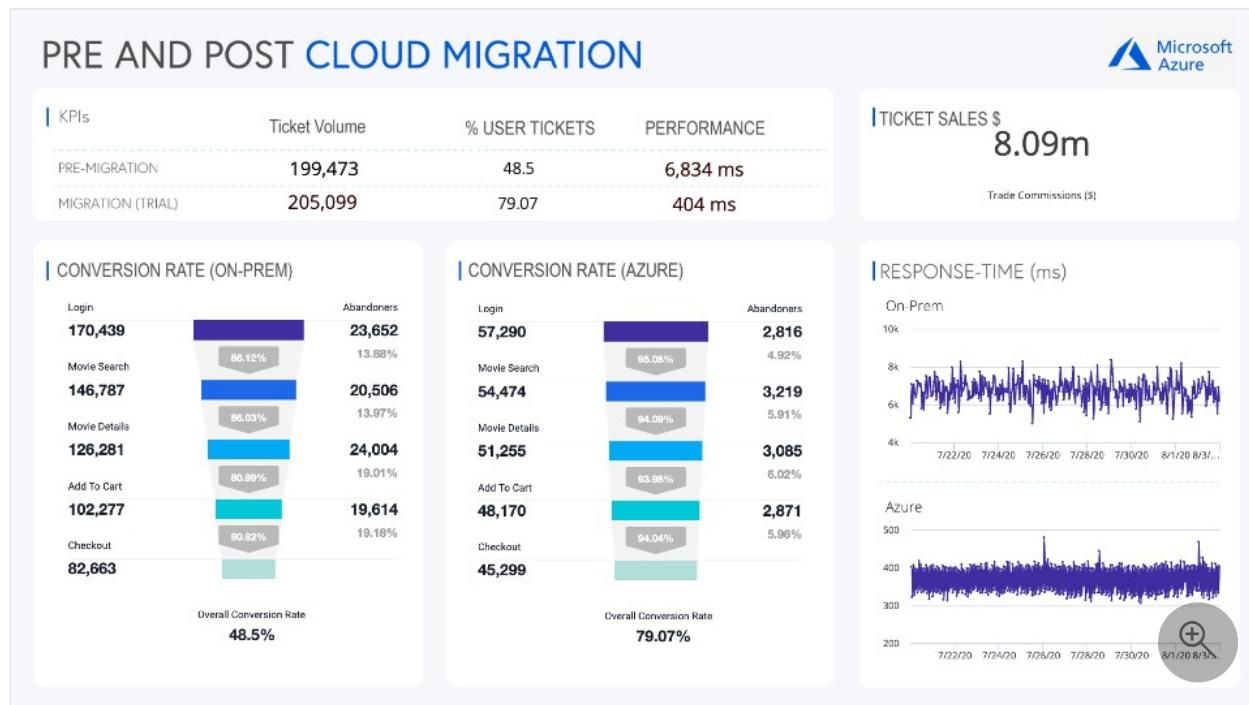


Figure 3: AppDynamics migration comparison.

Next steps

AppDynamics can help you measure business outcomes during your cloud adoption. For more information about using AppDynamics with Azure, see [Azure observability, right out of the box](#).

Initial organization alignment

Article • 04/28/2023

The most important aspect of any cloud adoption plan is the alignment of people who make the plan a reality. No plan is complete until you understand its people-related aspects.

Proper organizational alignment takes time. It becomes essential to establish long-term organizational alignment, especially as cloud adoption scales across the business and IT culture. Alignment is so important that there's an entire section dedicated to it in the [Organize methodology](#) of the Cloud Adoption Framework.

Full organization alignment isn't a required component of the cloud adoption plan. However, some initial organizational alignment is needed. This article outlines a best-practice starting point for organizational alignment. The guidance here can help complete your plan and prepare your teams for cloud adoption. When ready, you can use the [organization alignment](#) section to customize this guidance to fit your organization.

Initial best-practice structure

To create a balance between speed and control, we recommend that during cloud adoption, at a minimum, you have people accountable for *cloud adoption* and *cloud governance*. This might be a team of people sharing responsibilities for these areas or *capabilities*. It might also be individual people who are both accountable for the outcomes and responsible for the work. In either scenario, cloud adoption and cloud governance are two capabilities that involve natural friction between moving quickly and reducing risks. Here's how the two teams fit together:



Cloud adoption requires people to execute the cloud adoption tasks, and therefore few people are surprised that a cloud adoption team is needed. However, those new to the cloud may not fully appreciate the importance of a cloud governance team. This challenge often occurs early in adoption cycles. The cloud governance team provides the necessary checks and balances to ensure that cloud adoption doesn't expose the

business to new risks. When risks must be taken, this team ensures that proper processes and controls are implemented to mitigate or govern those risks.

One of the capabilities many newcomers to the cloud tend to overlook is the ability to drive greater velocity through *automation*. As you start your adoption projects, start building your automation muscle. Building automation frees up time, drive greater consistency, and demonstrate the model of the cloud that all parts of the business can benefit from.

Imagine a business user being able to self-serve and create cloud resources for a specific task without prior knowledge. Automation doesn't need to be overly complex at the beginning. It could be simple tasks, such as building VMs, storage accounts, or WebApps. Over time, these can evolve into more complex tasks. The Automation Function can enable this velocity by automating many repeatable tasks. See the [automation function](#) in the Organize section.

For more information about cloud adoption, cloud governance, and other such capabilities, see the brief section on [understanding required cloud capabilities](#).

Map people to capabilities

Assuming the suggested structure aligns with your cloud adoption plan, the next step is to map specific people to the necessary capabilities. To do so, answer the following questions:

- What person (or group of people) will be responsible for completing technical tasks in the cloud adoption plan?
- Who will be accountable for the team's ability to deliver technical changes?
- What person (or group of people) will be responsible for implementing protective governance mechanisms?
- What person is accountable for defining those governance controls?
- What person (or group of people) will be responsible for planning the sustainability and cloud efficiency?
- Are there other capabilities or people with accountability or responsibility within the cloud adoption plan?

After documenting the answers to these questions, you can establish [plans for skills readiness](#) to define plans to prepare these people for future work.

Evolving your organization structure

Your cloud adoption evolves, and your organization needs to keep up. Many critical functions need to be addressed, and while they don't need to be handled by dedicated teams, you need to cover these functions and the tasks.

Thinking through your organization, you want to ensure that you have functions that are dedicated to *Strategy*. This group is responsible for driving direction and business alignment.

- *Cloud Center of Excellence* will ensure that you're addressing the cultural change, skills, and systems needed to help build your cloud competency
- *Governance* defining controls to manage risk
- *Automation* to deliver consistency through code
- *Platform* ensuring that the environment is well-maintained and secure
- *Adoption* building of workloads and applications in a well-architected manner
- *Operations* ensuring that day-to-day issues are dealt with swiftly.

Depending on the organization's size, there might be dedicated *Security* and *Data* functions. These groups handle all aspects of security across the business and define rules for managing and governing organizational data.

In the [Organize](#) section, we outline the functions, the cadence of meetings, the deliverables, and typical sources of skills and traditional roles that can fit the function.

Introduce sustainability

Many organizations are evolving their organizational structure by introducing sustainability practices within the company. For example, modern Cloud Center of Excellence teams can encompass the directions for setting up the sustainability practice. Meanwhile, the governance team might define the accountabilities to ensure sustainability targets are being tracked and measured in alignment with the [organizational objectives and key results](#).

For additional considerations in defining roles, responsibilities, and standards, see [Plan for sustainability - Roles and responsibilities](#).

Next steps

Learn how to plan for cloud adoption.

[Plan for cloud adoption](#)

Adapt existing roles, skills, and processes for the cloud

Article • 02/28/2023

At each phase of the IT industry's history, the most notable changes have often been marked by changes in staff roles. One example is the transition from mainframe computing to client/server computing. The role of the computer operator during this transition has largely disappeared, replaced by the system administrator role. When virtualization arrived, the requirement for individuals working with physical servers was replaced with a need for virtualization specialists.

Roles will likely change as institutions similarly shift to cloud computing. For example, datacenter specialists might be replaced with cloud administrators or cloud architects. In some cases, though IT job titles haven't changed, the daily work of these roles has changed significantly.

IT staff members might feel anxious about their roles and positions because they realize that they need a different set of skills to support cloud solutions. But agile employees who explore and learn new cloud technologies shouldn't fear. They can lead the adoption of cloud services and help the organization learn and embrace the associated changes.

For guidance on building a new skill set, see the [skills readiness path](#).

Capture concerns

As the organization prepares for a cloud adoption effort, each team should document staff concerns as they arise by identifying:

- The type of concern. For example, workers might be resistant to the changes in job duties that come with the adoption effort.
- The impact if the concern isn't addressed. For example, resistance to adoption might result in workers being slow to execute the required changes.
- The area equipped to address the concern. For example, if workers in the IT department are reluctant to acquire new skills, the IT stakeholder's area is best equipped to address this concern. Identifying the area might be clear for some concerns. In these cases, you might need to escalate to executive leadership.

IT staff members commonly have concerns about acquiring the training needed to support expanded functions and new duties. Learning the training preferences of the

team helps you prepare a plan. It also allows you to address these concerns.

Identify gaps

Identifying gaps is another important aspect of organization readiness. A *gap* is a role, skill, or process that is required for your digital transformation but doesn't currently exist in your enterprise.

1. Enumerate the responsibilities that come with the digital transformation.
Emphasize new responsibilities and existing responsibilities to be retired.
2. Identify the area that aligns with each responsibility. For each new responsibility, check how closely it aligns with the area. Some responsibilities might span several areas. This crossover represents an opportunity for better alignment that you should document as a concern. In the case where no area is identified as being responsible, document this gap.
3. Identify the skills necessary to support each responsibility, and check if your enterprise has existing resources with those skills. Where there are no existing resources, determine the training programs or talent acquisition necessary to fill the gaps. Also determine the deadline by which you must support each responsibility to keep your digital transformation on schedule.
4. Identify the roles that will execute these skills. Some of your existing workforce will assume parts of the roles. In other cases, entirely new roles might be necessary.

Partner across teams

The skills necessary to fill the gaps in your organization's digital transformation are typically not confined to a single role or even a single department. Skills will have relationships and dependencies that can span a single role or multiple roles. Those roles might exist in several departments. For example, a workload owner might require someone in an IT role to provision core resources like subscriptions and resource groups.

These dependencies represent new processes that your organization implements to manage the workflow among roles. The preceding example shows several types of processes that support the relationship between the workload owner and the IT role. For instance, you can create a workflow tool to manage the process or use an email template.

Track these dependencies and make note of the processes that will support them. Also note whether the processes currently exist. For processes that require tooling, ensure

that the timeline for deploying any tools aligns with the overall digital-transformation schedule.

Next steps

Ensuring proper support for the translated roles is a team effort. To act on this guidance, review the organizational readiness overview to identify the right team structures and participants.

Identify the right team structures

Get started on a skills readiness path

Article • 08/12/2024

As more organizations embrace cloud solutions, IT professionals might feel anxious about their roles. Cloud technologies require a different set of skills. However, learning these new skills doesn't have to be hard. With the right training, you can gain the expertise and confidence to help your organization understand and embrace the changing technology landscape.

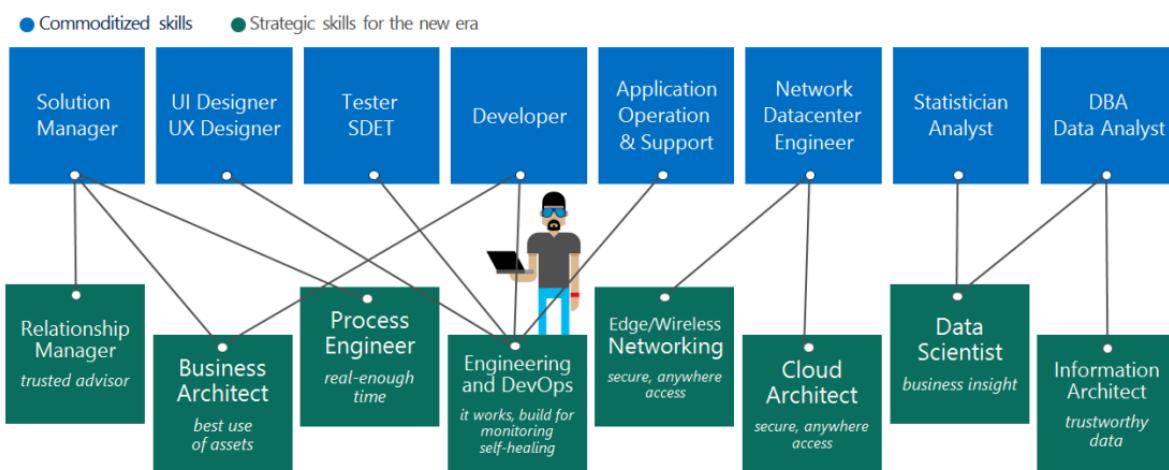


Figure 1: Mapping of skills to IT roles in a cloud-hosted environment.

Cloud Adoption Framework resources

The Cloud Adoption Framework guides you through the full adoption lifecycle. Each section of the framework provides opportunities to build necessary skills. To help you get started, the following skills-readiness paths describe key skills for success in adoption phases.

- **Strategy phase:** Develop the skills needed to prepare an actionable migration plan, including business justification and other required business-planning skills.
- **Readiness phase:** Develop the skills needed to prepare the business, culture, people, and environment for coming changes.
- **Migration phase:** Gain the skills required to implement the cloud migration plan.
- **Management phase - Monitoring:** Gain the skills needed to monitor a cloud environment.

Each of these learning paths shares opportunities across multiple media types to maximize knowledge acquisition.

You can also explore certification paths. For more information, see the [Certification process overview](#).

Microsoft Learn

[Microsoft Learn](#) can help you develop valuable IT skills at your own pace. With hands-on learning and customizable paths, Microsoft training offers a rewarding approach to online training. Earn points and levels, and achieve more!

Examples of tailored learning paths that align to the Cloud Adoption Framework include:

- [Evolve your DevOps practices](#): DevOps is the union of people, process, and products to enable continuous delivery of value to your end users. Azure DevOps is a set of services that gives you the tools you need to do just that. With Azure DevOps, you can build, test, and deploy any application, either in the cloud or on premises.
- [Azure for the data engineer](#): Explore how the world of data has evolved and how the advent of cloud technologies provides new opportunities for business to explore. You'll learn about various data platform technologies, and how a data engineer can take advantage of this technology to an organization's benefit.
- [The Principles of Sustainable Software Engineering](#): Sustainability, or cloud efficiency, is a fundamental topic to understand to optimize your cloud infrastructure, applications, and workloads. Regarding skilling and sustainability, you might benefit from reading the report on [Closing the Sustainability Skills Gap: Helping businesses move from pledges to progress ↗](#).

Learn more

To discover more learning paths, browse the [Microsoft Learn training catalog](#). Use the roles filter to align learning paths with your role.

Learn about paths to [Microsoft Certifications](#).

Feedback

Was this page helpful?



Plan for cloud adoption

Article • 02/28/2023

A plan is an essential requirement for a successful cloud adoption. A cloud adoption plan is an iterative project plan that helps a company transition from traditional IT approaches to transformation over to modern, agile approaches. This article series outlines how a cloud adoption plan helps companies balance their IT portfolio and manage transitions over time. Through this process, business objectives can be clearly translated into tangible technical efforts. Those efforts can then be managed and communicated in ways that make sense to business stakeholders. However, adopting such a process may require some changes to traditional project-management approaches.

Align strategy and planning

Cloud adoption plans start with a well-defined strategy. At a minimum, the strategy should outline the motivations, business outcomes, and business justifications for cloud adoption. Those positive returns are then balanced by the effort required to realize them.

The effort starts with the digital estate (proposed or existing), which translates the strategy into more tangible workloads and assets. You can then map these tangible elements to technical work. From there, skilled people in a proper organizational structure can execute the technical work. The cloud adoption plan combines these topics into one plan that can be forecasted, budgeted, implemented, and managed by means of agile project-management practices. This article series helps you build the plan and provides a few templates to make the job easier.

Transition from sequential to iterative planning

Planning for cloud adoption can be a significant change for some organizations. IT organizations have long focused on the application of linear or sequential models of project management, like the [waterfall model](#). In traditional IT, this approach was entirely logical. Most large IT projects started with a procurement request to acquire expensive hardware resources. Capital expense requests, budget allocations, and equipment acquisition often represented a large percentage of project execution. And after it was acquired, the hardware itself became a constraint on what could be delivered.

The acquisition models of the cloud change the core dependencies that made a sequential model necessary. The replacement of acquisition cycles with an operating-expense approach helps businesses move more quickly and with smaller financial commitments. This approach helps teams to engage in projects before all requirements are well known. It also creates room for a growth mindset, which frees the team to experiment, learn, and deliver without artificial constraints. For all these reasons and more, we highly recommend that teams use agile or iterative approaches to cloud adoption planning.

Build your cloud adoption plan

This article series walks through each step of translating strategy and effort into an actionable cloud adoption plan:

1. **Prerequisites:** Confirm that all prerequisite steps have been completed before you create your plan.
2. **Define and prioritize workloads:** Prioritize your first 10 workloads to establish an initial adoption backlog.
3. **Align assets to workloads:** Identify which assets (proposed or existing) are required to support the prioritized workloads.
4. **Review rationalization decisions:** Review rationalization decisions to refine adoption path decisions: migrate or innovate.
5. **Establish iterations and release plans:** *Iterations* are the time blocks allocated to do work. *Releases* are the definition of the work to be done before triggering a change to production processes.
6. **Estimate timelines:** Establish rough timelines for release planning purposes, based on initial estimates.

Next steps

Before building your cloud adoption plan, ensure that all [necessary prerequisites](#) are in place.

[Review prerequisites](#)

Prerequisites for an effective cloud adoption plan

Article • 02/28/2023

A plan is only as effective as the data that's put into it. For a cloud adoption plan to be effective, there are two categories of input: *strategic* and *tactical*. The following sections outline the minimum data points required in each category.

Strategic inputs

Accurate strategic inputs ensure that the work being done contributes to achievement of business outcomes. The [strategy section of the Cloud Adoption Framework](#) provides a series of exercises to develop a clear strategy. The outputs of those exercises feed the cloud adoption plan. Before developing the plan, ensure that the following items are well defined as a result of those exercises:

- **Clear motivations:** Why are we adopting the cloud?
- **Defined business outcomes:** What results do we expect to see from adopting the cloud?
- **Business justification:** How will the business measure success?

Every member of the team that implements the cloud adoption plan should be able to answer these three strategic questions. Managers and leaders who are accountable for implementation of the plan should understand the metrics behind each question and any progress toward realizing those metrics.

Tactical inputs

Accurate tactical inputs ensure that the work can be planned accurately and managed effectively. The [plan section of the Cloud Adoption Framework](#) provides a series of exercises to develop planning artifacts before you develop your plan. These artifacts provide answers to the following questions:

- **Digital estate rationalization:** What are the top 10 priority workloads in the adoption plan? How many additional workloads are likely to be in the plan? How many assets are being considered as candidates for cloud adoption? Are the initial efforts focused more on migration or innovation activities?
- **Organization alignment:** Who will do the technical work in the adoption plan? Who is accountable for adherence to governance and compliance requirements?

- **Skills readiness:** How many people are allocated to perform the required tasks? How well are their skills aligned to cloud adoption efforts? Are partners aligned to support the technical implementation?

These questions are essential to the accuracy of the cloud adoption plan. At a minimum, the questions about digital estate rationalization must be answered to create a plan. To provide accurate timelines, the questions about organization and skills are also important.

Next steps

Define your cloud adoption plan by deploying the template to Azure DevOps Services.

[Define your cloud adoption plan using the template](#)

Cloud adoption plan and Azure DevOps

Article • 05/08/2023

Azure DevOps is a set of cloud-based tools for Azure customers who manage iterative projects. It also includes tools for managing deployment pipelines and other important aspects of DevOps.

In this article, you learn how to quickly deploy a backlog by using a template. The template aligns cloud adoption efforts to a standardized process. This process is based on guidance in the Cloud Adoption Framework.

Create your cloud adoption plan

To deploy the cloud adoption plan, open the [Azure DevOps demo generator](#). This tool deploys the template to your Azure DevOps tenant. This tool requires the following steps:

1. Verify that the **Selected Template** field is set to **Cloud Adoption Plan**. If it isn't, select **Choose template** to choose the right template.
2. Select your Azure DevOps organization from the **Select Organization** menu.
3. Type a name for your new project. The cloud adoption plan has this name when it's deployed to your Azure DevOps tenant.
4. Select **Create Project** to create a project in your tenant that's based on the strategy and plan template. A progress bar shows your progress toward the deployment of the project.
5. When deployment is finished, select **Navigate to project** to see your new project.

After your project has been created, continue through this article series to learn how to modify the template to align to your cloud adoption plan.

For more support and guidance on this tool, see [Azure DevOps Services demo generator](#).

Bulk edit the cloud adoption plan

After you deploy your project plan, you can use Microsoft Excel to modify it. It's easier to create workloads or assets by using Microsoft Excel compared to the Azure DevOps browser experience.

To prepare your workstation for bulk editing, see [Bulk add or modify work items with Microsoft Excel](#).

Use the cloud adoption plan

The cloud adoption plan organizes activities by activity type:

- **Epics:** An epic represents an overall phase of the cloud adoption lifecycle.
- **Features:** Features are used to organize specific objectives within each phase. For instance, migration of a specific workload is a feature.
- **User stories:** User stories group work into logical collections of activities based on a specific goal.
- **Tasks:** Tasks are the actual work to be done.

At each layer, activities are sequenced based on dependencies. Activities are linked to articles in the Cloud Adoption Framework to clarify the objective or task at hand.

The clearest view of the cloud adoption plan comes from the **Epics** backlog view. For help with the backlog view, see [view a backlog](#). From the backlog view, it's easy to plan and manage the work required to complete the current phase of the adoption lifecycle.

ⓘ Note

The current state of the cloud adoption plan focuses heavily on migration efforts. Tasks related to governance, innovation, or operations must be populated manually.

Align the cloud adoption plan

The overview pages for the Strategy methodology and the Plan methodology refer to the [strategy and plan template](#). The template organizes the decisions and data points. The data points align the template for the cloud adoption plan with your specific plans for adoption. Consider completing the exercises in the [Strategy methodology](#) and the [Plan methodology](#) before your new project begins.

The following articles support alignment of the cloud adoption plan:

- **Workloads:** To capture workloads that might be migrated or modernized, align features within the cloud migration epic. Add and modify those features to migrate your top 10 workloads.
- **Assets:** Each asset (virtual machine, application, or data) is represented by the user stories under each workload. Add and modify those user stories to align with your digital estate.

- **Rationalization:** As each workload is defined, the initial assumptions about that workload can be challenged. The workload might result in changes to the tasks under each asset.
- **Create release plans:** Iteration paths establish release plans by aligning efforts with various releases and iterations.
- **Establish timelines:** To define the start and end dates for each iteration, create a timeline to manage the overall project.

These five articles help with the alignment tasks required to start managing your adoption efforts. The next step gets you started on the alignment exercise.

Next steps

Start aligning your plan project by [defining and prioritizing workloads](#).

[Define and prioritize workloads](#)

Define and prioritize workloads for a cloud adoption plan

Article • 04/28/2023

Establishing clear, actionable priorities is one of the secrets to successful cloud adoption. The natural temptation is to invest time defining all workloads that could be affected during cloud adoption. But that's counterproductive, especially early in the adoption process.

Instead, we recommend that your team thoroughly prioritizes and documents the first 10 workloads. After implementing the adoption plan, the team can maintain a list of the following 10 highest-priority workloads. This approach provides enough information to plan for the subsequent few iterations.

Limiting the plan to 10 workloads encourages agility and alignment of priorities as business criteria change. This approach also allows the cloud adoption team to learn and refine estimates. Most importantly, it removes extensive planning as a barrier to effective business change.

What is a workload?

In cloud adoption, a workload is a collection of IT assets (servers, VMs, applications, data, or appliances) that collectively support a defined process. Workloads can support more than one process. Workloads can also depend on other shared assets or larger platforms. However, a workload should have defined boundaries regarding the dependent assets and the processes that depend upon the workload. Often, workloads can be visualized by monitoring network traffic among IT assets.

Prerequisites

The strategic inputs from the prerequisites list make accomplishing the following tasks much more manageable. For help with gathering the data discussed in this article, review the [prerequisites](#).

Initial workload prioritization

During the process of [incremental rationalization](#), your team should agree on a [Power of 10 approach](#) consisting of 10 priority workloads. These workloads serve as an initial boundary for adoption planning.

Suppose you decide that a digital estate rationalization isn't needed. In that case, we recommend that the cloud adoption teams and the cloud strategy team agree on a list of 10 applications to serve as the initial focus of the migration. We recommend that these 10 workloads contain a mixture of simple workloads (fewer than 10 assets in a self-contained deployment) and more complex workloads. Those 10 workloads start the workload prioritization process.

Note

The Power of 10 approach serves as an initial planning boundary, focusing the energy and investment in early-stage analysis. However, analyzing and defining workloads is likely to cause changes in the list of priority workloads.

Add workloads to your cloud adoption plan

In the previous article, [Cloud adoption plan and Azure DevOps](#), you created a cloud adoption plan in Azure DevOps.

You can now represent the workloads in the Power of 10 list in your cloud adoption plan. The easiest way to do this is via bulk editing in Microsoft Excel. To prepare your workstation for bulk editing, see [Bulk add or modify work items with Microsoft Excel](#).

Step 5 in that article tells you to select **Input list**. Instead, select **Query list**. Then, from the **Select a Query** drop-down list, select the **Workload Template** query. That query loads all the efforts related to migrating a single workload into your spreadsheet.

After the work items for the workload template are loaded, follow these steps to begin adding new workloads:

1. Copy all items with the **Workload Template** tag in the far right column.
2. Paste the copied rows below the last line item in the table.
3. Change the title cell for the new feature from **Workload Template** to the name of your new workload.
4. Paste the new workload name cell into the tag column for all rows below the new feature. Be careful not to change the tags or name of the rows related to the actual **Workload Template** feature. When you add the next workload to the cloud adoption plan, you'll need those work items.
5. Skip to step 8 in the bulk-editing instructions to publish the worksheet. This step creates all the work items required to migrate your workload.

Repeat steps 1 through 5 for each workload in the Power of 10 list.

Define workloads

After defining initial priorities and adding workloads to the plan, define each workload via deeper qualitative analysis. Before including any workload in the cloud adoption plan, try to provide the following data points for each workload.

Business inputs

Data point	Description	Input
Workload name	What is this workload called?	
Workload description	In one sentence, what does this workload do?	
Adoption motivations	Which cloud adoption motivations are affected by this workload?	
Primary sponsor	Of those stakeholders affected, who is the primary sponsor requesting the preceding motivations?	
Business impact	What is the business impact of this workload?	
Application impact	What impact does this application have on business processes?	
Data impact	What impact does the data have on the business?	
Business unit	Which business unit is responsible for the cost of this workload?	
Business processes	Which business processes will be affected by changes to the workload?	
Business teams	Which business teams will be affected by changes?	
Business stakeholders	Are there any executives whose business will be affected by changes?	
Business outcomes	How will the business measure the success of this effort?	
Metrics	What metrics will be used to track success?	
Compliance	Are there any third-party compliance requirements for this workload?	
Application owners	Who is accountable for the business impact of any applications associated with this workload?	

Data point	Description	Input
Business freeze periods	Are there any times when the business won't permit change?	
Geographies	Are any geographies affected by this workload?	
Sustainability	What sustainability and cloud efficiency considerations have been taken into account for this workload?	

Technical inputs

Data point	Description	Input
Adoption approach	Is this adoption a candidate for migration or innovation?	
Application ops lead	List the parties responsible for the performance and availability of this workload.	
SLAs	List any service-level agreements (RTO/RPO requirements).	
Criticality	List the current application criticality.	
Data classification	List the classification of data sensitivity.	
Operating geographies	List any geographies in which the workload is or should be hosted.	
Applications	Specify an initial list or count of any applications included in this workload.	
VMs	Specify an initial list or count of any VMs or servers included in the workload.	
Data sources	Specify an initial list or count of any data sources included in the workload.	
Dependencies	List any asset dependencies not included in the workload.	
User traffic geographies	List geographies with a significant collection of user traffic.	

Confirm priorities

Based on the assembled data, the cloud strategy and adoption teams should meet to reevaluate priorities. Clarification of business data points might prompt changes in priorities. Technical complexity or dependencies might result in changes related to staffing allocations, timelines, or sequencing of technical efforts.

After a review, both teams should be comfortable with confirming the resulting priorities. This set of documented, validated, and confirmed priorities is the prioritized cloud adoption backlog.

Next steps

For any workload in the prioritized cloud adoption backlog, the team is now ready to align assets.

[Align assets for prioritized workloads](#)

Align assets to prioritized workloads

Article • 02/28/2023

Workload is a conceptual description of a collection of assets: VMs, applications, and data sources. The previous article, [Define and prioritize](#), provided guidance for collecting the data that will define the workload. Before migration, a few of the technical inputs in that list require additional validation. This article helps with validation of the following inputs:

- **Applications:** List any applications included in this workload.
- **VMs and servers:** List any VMs or servers included in the workload.
- **Data sources:** List any data sources included in the workload.
- **Dependencies:** List any asset dependencies not included in the workload.

There are several options for assembling this data. The following are a few of the most common approaches.

Alternative inputs: Migrate, modernize, innovate

The objective of the preceding data points is to capture relative technical effort and dependencies as an aid to prioritization. Depending on the transition you want, you may need to gather alternative data points to support proper prioritization.

Migrate: For pure migration efforts, the existing inventory and asset dependencies serve as a fair measure of relative complexity.

Modernize: When the goal for a workload is to modernize applications or other assets, these data points are still solid measures of complexity. However, it might be wise to add an input for modernization opportunities to the workload documentation.

Innovate: When data or business logic is undergoing material change during a cloud adoption effort, it's considered an *innovate* type of transformation. The same is true when you're creating new data or new business logic. For any innovate scenarios, the migration of assets will likely represent the smallest amount of effort required. For these scenarios, the team should devise a set of technical data inputs to measure relative complexity.

Azure Migrate

Azure Migrate provides a set of grouping functions that can speed up the aggregation of applications, VMs, data sources, and dependencies. After workloads have been defined conceptually, they can be used as the basis for grouping assets based on dependency mapping.

The Azure Migrate documentation provides guidance on [how to group machines based on dependencies](#).

Configuration-management database

Some organizations have a well-maintained configuration-management database (CMDB) within their existing operations-management tooling. They could use the CMDB alternatively to provide the input data points discussed earlier.

Next steps

Review [rationalization decisions](#) based on asset alignment and workload definitions.

[Review rationalization decisions](#)

Review rationalization decisions

Article • 02/28/2023

During initial strategy and planning stages, we suggest you apply an [incremental rationalization](#) approach to the digital estate. But this approach embeds some assumptions into the resulting decisions. We advise the cloud strategy team and the cloud adoption teams to review those decisions in light of expanded-workload documentation. This review is also a good time to involve business stakeholders and the executive sponsor in future state decisions.

Important

Further validation of the rationalization decisions will occur during the assessment phase of migration. This validation focuses on business review of the rationalization to align resources appropriately.

To validate rationalization decisions, use the following questions to facilitate a conversation with the business. The questions are grouped by the likely rationalization alignment.

Innovation indicators

If the joint review of the following questions yields an affirmative answer, a workload might be a better candidate for innovation. Such a workload wouldn't be migrated via a lift and shift or modernize model. Instead, the business logic or data structures would be re-created as a new or rearchitected application. This approach can be more labor-intensive and time-consuming. But for a workload that represents significant business returns, the investment is justified.

- Do the applications in this workload create market differentiation?
- Is there a proposed or approved investment aimed at improving the experiences associated with the applications in this workload?
- Does the data in this workload make new product or service offerings available?
- Is there a proposed or approved investment aimed at taking advantage of the data associated with this workload?
- Can the effect of the market differentiation or new offerings be quantified? If so, does that return justify the increased cost of innovation during cloud adoption?

The following two questions can help you include high-level technical scenarios in the rationalization review. Answering "yes" to either could identify ways of accounting for or

reducing the cost associated with innovation.

- Will the data structures or business logic change during the course of cloud adoption?
- Is an existing deployment pipeline used to deploy this workload to production?

If the answer to either question is "yes," the team should consider including this workload as an innovation candidate. At a minimum, the team should flag this workload for architecture review to identify modernization opportunities.

Migration indicators

Migration is a faster and cheaper way of adopting the cloud. But it doesn't take advantage of opportunities to innovate. Before you invest in innovation, answer the following questions. They can help you determine whether a migration model is more applicable for a workload.

- Is the source code supporting this application stable? Do you expect it to remain stable and unchanged during the time frame of this release cycle?
- Does this workload support production business processes today? Will it do so throughout the course of this release cycle?
- Is it a priority that this cloud adoption effort improves the stability and performance of this workload?
- Is cost reduction associated with this workload an objective during this effort?
- Is reducing operational complexity for this workload a goal during this effort?
- Is innovation limited by the current architecture or IT operation processes?

If the answer to any of these questions is "yes," you should consider a migration model for this workload. This recommendation is true even if the workload is a candidate for innovation.

Challenges in operational complexity, costs, performance, or stability can hinder business returns. You can use the cloud to quickly produce improvements related to those challenges. Where it's applicable, we suggest you use the migration approach to first stabilize the workload. Then expand on innovation opportunities in the stable, agile cloud environment. This approach provides short-term returns and reduces the cost required to drive long-term change.

Important

Migration models include incremental modernization. Using platform as a service (PaaS) architectures is a common aspect of migration activities. So too are minor

configuration changes that use those platform services. The boundary for migration is defined as a material change to the business logic or supporting business structures. Such change is considered an innovation effort.

Update the project plan

The skills required for a migration effort are different from the skills required for an innovation effort. During implementation of a cloud adoption plan, we suggest that you assign migration and innovation efforts to different teams. Each team has its own iteration, release, and planning cadences. Assigning separate teams provides the process flexibility to maintain one cloud adoption plan while accounting for innovation and migration efforts.

When you manage the cloud adoption plan in Azure DevOps, that management is reflected by changing the parent work item (or epic) from cloud migration to cloud innovation. This subtle change helps ensure all participants in the cloud adoption plan can quickly track the required effort and changes to remediation efforts. This tracking also helps align proper assignments to the relevant cloud adoption team.

For large, complex adoption plans with multiple distinct projects, consider updating the iteration path. Changing the area path makes the workload visible only to the team assigned to that area path. This change can make work easier for the cloud adoption team by reducing the number of visible tasks. But it adds complexity for the project management processes.

Next steps

[Establish iterations and release plans](#) to begin planning work.

[Establish iterations and release plans](#) to begin planning work.

Establish iterations and release plans

Article • 02/28/2023

Agile and other iterative methodologies are built on the concepts of iterations and releases. This article outlines the assignment of iterations and releases during planning. Those assignments drive timeline visibility to make conversations easier among members of the cloud strategy team. The assignments also align technical tasks in a way that the cloud adoption team can manage during implementation.

Establish iterations

In an iterative approach to technical implementation, you plan technical efforts around recurring time blocks. Iterations tend to be one-week to six-week time blocks. Consensus suggests that two weeks is the average iteration duration for most cloud adoption teams. But the choice of iteration duration depends on the type of technical effort, the administrative overhead, and the team's preference.

To begin aligning efforts to a timeline, we suggest that you define a set of iterations that last 6 to 12 months.

Understand velocity

Aligning efforts to iterations and releases requires an understanding of velocity. Velocity is the amount of work that can be completed in any given iteration. During early planning, velocity is an estimate. After several iterations, velocity becomes a highly valuable indicator of the commitments that the team can make confidently.

You can measure velocity in abstract terms like story points. You can also measure it in more tangible terms like hours. For most iterative frameworks, we recommend using abstract measurements to avoid challenges in precision and perception. Examples in this article represent velocity in hours per sprint. This representation makes the topic more universally understood.

Example: A five-person cloud adoption team has committed to two-week sprints. Given current obligations like meetings and support of other processes, each team member can consistently contribute 20 hours per week to the adoption effort. For this team, the initial velocity estimate is 100 hours per sprint.

Iteration planning

Initially, you plan iterations by evaluating the technical tasks based on the prioritized backlog. Cloud adoption teams estimate the effort required to complete various tasks. Those tasks are then assigned to the first available iteration.

During iteration planning, the cloud adoption teams validate and refine estimates. They do so until they have aligned all available velocity to specific tasks. This process continues for each prioritized workload until all efforts align to a forecasted iteration.

In this process, the team validates the tasks assigned to the next sprint. The team updates its estimates based on the team's conversation about each task. The team then adds each estimated task to the next sprint until the available velocity is met. Finally, the team estimates additional tasks and adds them to the next iteration. The team performs these steps until the velocity of that iteration is also exhausted.

The preceding process continues until all tasks are assigned to an iteration.

Example: Let's build on the previous example. Assume each workload migration requires 40 tasks. Also assume you estimate each task to take an average of one hour. The combined estimation is approximately 40 hours per workload migration. If these estimates remain consistent for all 10 of the prioritized workloads, those workloads will take 400 hours.

The velocity defined in the previous example suggests that the migration of the first 10 workloads will take four iterations, which is two months of calendar time. The first iteration will consist of 100 tasks that result in the migration of two workloads. In the next iteration, a similar collection of 100 tasks will result in the migration of three workloads.

Warning

The preceding numbers of tasks and estimates are strictly used as an example. Technical tasks are seldom that consistent. You shouldn't see this example as a reflection of the amount of time required to migrate a workload.

Release planning

Within cloud adoption, a release is defined as a collection of deliverables that produce enough business value to justify the risk of disruption to business processes.

Releasing any workload-related changes into a production environment creates some changes to business processes. Ideally, these changes are seamless, and the business

sees the value of the changes with no significant disruptions to service. But the risk of business disruption is present with any change and shouldn't be taken lightly.

To ensure a change is justified by its potential return, the cloud strategy team should participate in release planning. Once tasks are aligned to sprints, the team can determine a rough timeline of when each workload will be ready for production release. The cloud strategy team would review the timing of each release. The team would then identify the inflection point between risk and business value.

Example: Continuing the previous example, the cloud strategy team has reviewed the iteration plan. The review identified two release points. During the second iteration, a total of five workloads will be ready for migration. Those five workloads will provide significant business value and will trigger the first release. The next release will come two iterations later, when the next five workloads are ready for release.

Assign iteration paths and tags

For customers who manage cloud adoption plans in Azure DevOps, the previous processes are reflected by assigning an iteration path to each task and user story. We also recommend tagging each workload with a specific release. That tagging and assignment feed the automatic population of timeline reports.

Next steps

[Estimate timelines](#) to properly communicate expectations.

[Estimate timelines](#)

Timelines in a cloud adoption plan

Article • 02/28/2023

In the previous article in this series, workloads and tasks were assigned to [releases and iterations](#). Those assignments feed the timeline estimates in this article.

Work breakdown structures are commonly used in sequential project-management tools. They represent how dependent tasks will be completed over time. Such structures work well when tasks are sequential in nature. The interdependencies in tasks found in cloud adoption make such structures difficult to manage. To fill this gap, you can estimate timelines based on iteration-path assignments by hiding complexity.

Estimate timelines

To develop a timeline, start with releases. Those release objectives create a target date for any business impact. Iterations aid in aligning those releases with specific time durations.

If more granular milestones are required in the timeline, use iteration assignment to indicate milestones. To do this assignment, assume that the last instance of a workload-related task can serve as the final milestone. Teams also commonly tag the final task as a milestone.

For any level of granularity, use the last day of the iteration as the date for each milestone. This ties completion of workload adoption to a specific date. You can track the date in a spreadsheet or a sequential project-management tool like Microsoft Project.

Delivery plans in Azure DevOps

If you're using Azure DevOps to manage your cloud adoption plan, consider using the [Microsoft Delivery Plans extension](#) . This extension can quickly create a visual representation of the timeline that is based on iteration and release assignments.

Plan a data warehouse migration

Article • 02/28/2023

A data warehouse migration is a challenge for any company. In order to execute it well and avoid any unwelcome surprises and unplanned costs, you need to thoroughly research the challenge, mitigate risk, and plan your migration to ensure that you're as ready as possible. At a high level, your plan should cover the core data warehouse migration process steps and any tasks within them. The main process steps are:

- Pre-migration preparation
- Migration strategy and execution
- Post-migration

For example, preparation includes things like readying your data warehouse migration team in terms of skills training and technology familiarization. It also includes setting up a proof of concept lab, understanding how you will manage test and production environments, gaining appropriate clearance to migrate your data and a production system outside of the corporate firewall and setting up migration software in your datacenter to enable migration to proceed.

For a data warehouse migration to proceed smoothly, your plan should establish a clear understanding of:

- Your business case, including its drivers, business benefits, and risks.
- Migration team roles and responsibilities.
- The skill set and training required to enable successful migration.
- Allocated budget for the complete migration.
- Your migration strategy.
- How you can avoid risk in the migration project to avoid delays or rework.
- Your existing data warehouse system, its architecture, schema, data volumes, data flows, security, and operational dependencies.
- Differences between your existing on-premises data warehouse DBMS and Azure Synapse, like data types, SQL functions, logic, and other considerations.
- What needs to be migrated and priorities.
- The migration tasks, approaches, order, and deadlines.
- How you will control migration.
- How to prevent user disruption while undertaking the migration.
- What you need to do on-premises to avoid delays and enable migration.
- Tools to enable secure migration of schemas, data, and ETL processing to Azure.
- Data model design changes that are required during and after migration.

- Any pre-migration or post-migration technology changes and how to minimize rework.
- Post-migration technology deprecation.
- How you will implement testing and quality assurance to prove success.
- Your checkpoints to assess progress and enable decisions to be made.
- Your contingency plan and points of rollback in case things go wrong.

In order to achieve this understanding, we need to prepare and begin specific activities before any migration starts. Let's look at what that entails in more detail.

Pre-migration preparation

There are several things that should be addressed before you even begin a data warehouse migration.

Key roles in a data warehouse migration team

Key roles in a migration project include:

- Business owner
- Project manager (with agile methodology experience such as Scrum)
- Project coordinator
- Cloud engineer
- Database administrator (existing data warehouse DBMS and Azure Synapse)
- Data modelers
- ETL developers
- Data virtualization specialist (possibly a database administrator)
- Testing engineer
- Business analysts (to help test BI tool queries, reports, and analyses)

In addition, the team need the support of your on-premises infrastructure team.

Skills and training to ready the team for migration

With respect to skills, expertise is important in a data warehouse migration. Therefore, ensure the appropriate members of your migration team are trained in Azure cloud fundamentals, Azure Blob storage, Azure Data Lake Storage, Azure Data Box, ExpressRoute, Azure identity management, Azure Data Factory, and Azure Synapse. Your data modelers will most likely need to fine-tune your Microsoft Azure Synapse data models once migration from your existing data warehouse has occurred.

Assessing your existing data warehouse

Another part of preparing to migrate is the need for a full assessment of your existing data warehouse to fully understand the architecture, data stores, schema, business logic, data flows, the DBMS functionality used, warehouse operation, and the dependencies. The more understanding is gained here the better. A detailed knowledge of how the system works helps to communicate and cover off all bases.

The purpose of the assessment is not just to ensure detailed understanding of the current setup across the migration team but also to understand strengths and weaknesses in the current setup. The outcome of an assessment of your current data warehouse therefore can impact your migration strategy in terms of lift and shift versus something broader. For example, if the outcome of an assessment is that your data warehouse is at end of life then clearly the strategy would be more of a data migration to a newly designed data warehouse on Azure Synapse versus a lift-and-shift approach.

On-premises preparation for data migration

In addition to preparing and readying your migration team for your target environment and assessing your current setup, it is equally important to also set things in motion on-premises as production data warehouses tend to be heavily controlled by IT procedures and approval processes. To avoid delays, ensure that your datacenter infrastructure and operations teams are ready for migrating your data, schema, ETL jobs, and so on, to the Azure cloud. Data migration can occur via:

- AzCopy to Azure Blob storage.
- Microsoft Azure ExpressRoute to transfer compressed data directly to Azure.
- File export to Azure Data Box.

The main factors influencing which of these options is selected are data volume size (in terabytes) and network speed (in Mbps). A calculation is needed to determine how long it would take to migrate the data via the network, considering that data might be compressed in your data warehouse and become uncompressed when you export it. This situation can slow data transfer. Recompress data via Gzip when moving data by any of the above methods. PolyBase can process Gzipped data directly. Large data volumes will likely be migrated via Azure Data Box if it will take too long to move the data.

Additionally, for Azure Data Factory to control the execution of exports of your existing data warehouse data from Azure, self-hosted integration run-time software must be installed in your datacenter to enable migration to proceed. Given these requirements if

formal approval is needed to make this possible, then starting the appropriate approval processes early to enable this to happen will help avoid delays down the line.

Azure preparation for schema and data migration

In terms of preparation on the Azure side, data import will need to be managed either via Microsoft Azure ExpressRoute or Microsoft Azure Data Box. Azure Data Factory pipelines are an ideal way to load your data into Azure Blob storage and then load from there into Azure Synapse using PolyBase. Therefore preparation is needed on the Azure side to develop such a pipeline.

The alternative is to use your existing ETL tool on Azure if it supports Azure Synapse, which means setting up the tool on Azure from Azure Marketplace and readying a pipeline to import your data and load it into Azure Blob storage.

Defining a migration strategy

Migration goals

In any strategy, there needs to be a set of objectives or goals that should be defined to indicate success. Targets can then be set to achieve these goals and people given responsibility for reaching them. Examples of migration goals and corresponding metrics to set targets for in a cloud data warehouse migration project are shown in the table below:

Types of goal and metric examples:

Improve overall performance

- Data migration performance
- ELT performance
- Data loading performance
- Complex query performance
- Number of concurrent users

Run at lower cost

- Cost of compute by workload, for example, number of compute hours x cost per hour for:
 - Standard reporting

- Ad hoc queries
- Batch ELT processing
- Cost of storage (staging, production tables, indexes, temporary space)

Operate with better availability and service levels

- Service-level agreements
- High availability

Improve productively

- Tasks automated, reduced administrative headcount

A successful data warehouse migration can therefore be interpreted as a data warehouse that runs as fast or faster and at lower cost than the legacy system you migrated from. Assigning owners of these goals creates accountability for reaching them. It also ensures that testing in a proof of concept lab (as defined in the de-risking section in this guide) will be deemed successful if the tests identify ways that the goals can be achieved.

Migration approach

You have several strategic options for migrating your existing data warehouse to Azure Synapse:

- Lift and shift your existing data warehouse as-is.
- Simplify your existing data warehouse and then migrate it.
- Completely redesign your data warehouse on Azure Synapse and migrate your data.

The findings of the assessment of your existing data warehouse should significantly influence your strategy. A good assessment outcome might recommend a lift and shift strategy. A mediocre outcome due to a low agility rating might indicate that simplification is needed before migration. A poor outcome might indicate a complete redesign is needed.

Lift and shift leaves your architecture as-is, trying to minimize the work in moving your existing system. If your existing ETL tool already supports Azure Synapse, you might be able to change the target with minimal effort. Nevertheless there will be differences in table types, data types, SQL functions, views, stored procedure business logic etc. These differences and ways around them are detailed in lower-level documents in this migration series.

Simplifying your existing data warehouse prior to migration is about reducing complexity to ease migration. It could include:

- Removing or archiving unused tables before migrating to avoid migrating data that is not used.
- Converting physical data marts to virtual data marts using data virtualization software to reduce what you have to migrate. Conversion also improves agility and reduces total cost of ownership, so it could be considered as modernization during migration.

You can also simplify first and then lift and shift what remains.

Migration scope

Whatever strategy you choose, you should clearly define the scope of the migration, what will be migrated, and whether you'll migrate incrementally or all at once. One example of incremental migration is migrating your data marts first, followed by your data warehouse. This approach would allow you to focus on high-priority business areas while allowing your team to incrementally build expertise as each mart is individually migrated, before migrating the data warehouse itself.

Defining what has to be migrated

Make an inventory of everything that needs to be migrated. This includes schema, data, ETL processes (pipelines), authorization privileges, users, BI tool semantic access layers, and analytic applications. A detail understanding of what's involved in migrating the inventory is provided in each of the lower-level migration articles in this series. Links to these are shown below.

- Schema migration, design, and performance considerations.
- Data migration, ETL processing, and load.
- Access security and data warehouse operations.
- Migration of visualization and reports.
- Minimizing the impact of SQL issues.
- Third-party tools to help you in your data warehouse migration.

If you are uncertain about the best approach, conduct tests in a proof of concept lab to identify optimal techniques. For more information, see the section on de-risking your data warehouse migration project.

Migration control

Data warehouse migration to Azure Synapse involves tasks that need to be conducted:

- On-premises, such as data export.
- On the network, such as data transfer.
- In the Azure cloud, such as data transformation, integration, and load.

The problem is that managing these tasks can be complicated if scripts and utilities are all being developed, tested, and run independently in both on-premises and Azure environments. It adds complexity especially if version control, test management and migration execution are not coordinated.

You should avoid these complexities and control them from a common place via a source control repository to manage change from development to testing and production. Migration execution will involve tasks that need to be performed on-premises, on the network, and in Azure. Because Azure Synapse is the target environment, controlling migration execution from Azure simplifies management. Use Azure Data Factory to create a migration control pipeline to control execution both on-premises and on Azure. This introduces automation and minimizes errors. Data Factory becomes a migration orchestration tool, not just an enterprise data integration tool.

Other options to control migration available from Microsoft partners running on Azure include data warehouse automation tools to try to automate migration. Vendors like WhereScape and Attunity for example. Most of these automation tools are aimed at a lift-and-shift approach to migration. Even then, there may be some things that may not be supported by such tools, for example, stored procedures. These products and several others are detailed in a separate guide dedicated to third-party tools to help you migrate to Azure Synapse.

Migration testing

The first thing you need for testing is to define a series of tests and a set of required outcomes for each test that need to be run to verify and indicate success. It is important to ensure that all aspects are tested and compared across your existing and migrated systems including:

- Schema
- Data types converted where necessary
- Use user-defined schema in Azure Synapse to distinguish between data warehouse and data mart tables
- Users
- Roles and assignments of users to those roles
- Data access security privileges

- Data privacy and compliance
- Privileges that govern administration capabilities
- Data quality and integrity
- ETL processing that populates Azure Synapse both into the data warehouse and from the data warehouse to any data marts, including testing
- All rows are correct in all tables including history
- Slowly changing dimension processing
- Change data capture processing
- Calculations and aggregations that use functions that could differ across systems
- Results of all known queries, reports, and dashboards
- Performance and scalability
- Analytical functionality
- Costs in the new pay-as-you-go environment

Automate testing as much as possible, making each test repeatable and enabling a consistent approach to evaluating results. If reports and dashboards are inconsistent, then having the ability to compare metadata lineage across original and migrated systems is valuable during migration testing, since it can highlight differences and pinpoint where they occurred when these are not easy to detect.

The best way to do this securely is to create roles, assign access privileges to roles and then attach users to roles. To access your newly migrated data warehouse, set up an automated process to create new users and assign roles. Do the same to remove users from roles.

Communicate the cutover to all users so they know what's changing and what to expect.

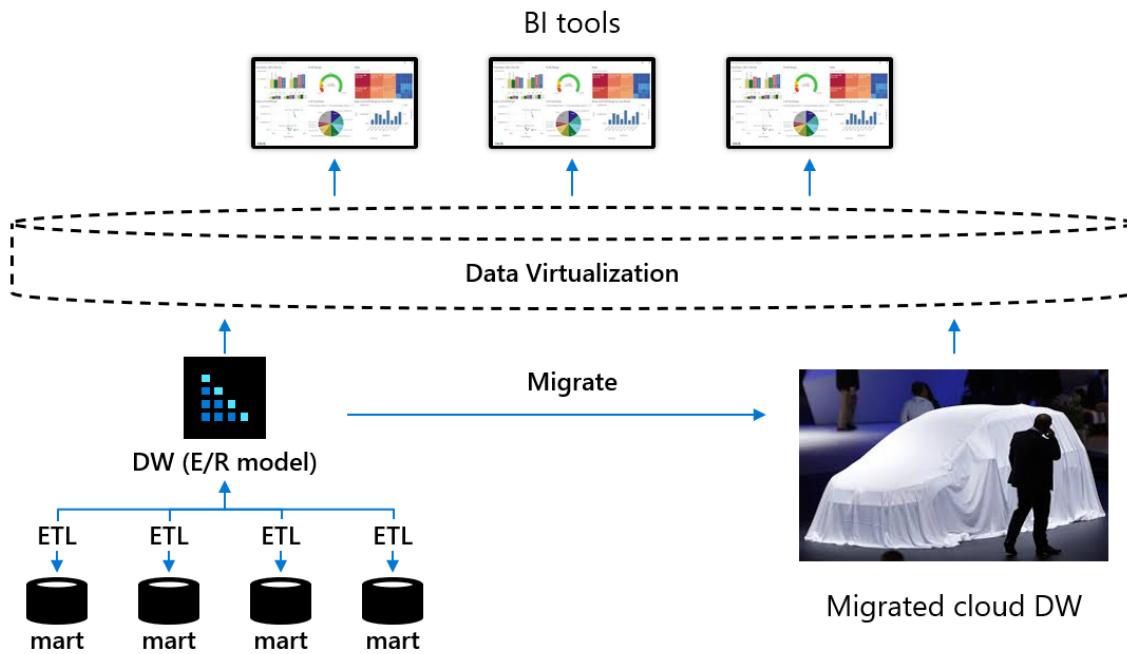
De-risking your data warehouse migration project

Another critical factor in data warehouse migration is de-risking the project in order to maximize the likelihood of a success. There are several things that can be done to de-risk a data warehouse migration. They include:

- Establishing a proof-of-concept lab to enable your team to try things, conduct tests, understand any issues and identify fixes and optimizations that help validate migration approaches, improve performance and lower costs. It also helps establish ways to automate tasks, use built-in tools and build templates to capture best practices, learn from experience, and track lessons learned. It's an invaluable way to mitigate risk and increase your chances of success. In addition, you can

assign owners to tests who are accountable for achieving migration goals and targets as defined in your migration strategy.

- Introduce data virtualization between BI tools and your data warehouse and data marts. Introduce user transparency using data virtualization to reduce risk in a data warehouse migration, and hide the migration from users by using data virtualization BI tools, as shown in the following diagram.



The purpose of this is to break the dependency between business users using self-service BI tools and the physical schema of the underlying data warehouse and data marts that are being migrated. By introducing data virtualization, any schema alterations made during data warehouse and data mart migration to Azure Synapse (for example, to optimize performance) can be hidden from business users because they only access virtual tables in the data virtualization layer. If structural change is needed, only the mappings between the data warehouse or data marts and any virtual tables would need to be changed so that users remain unaware of those changes and unaware of the migration.

- Look to archive any existing tables identified as never used prior to data warehouse migration as there is little point migrating tables that are not used. One possible way of doing this is to archive the unused data to Azure Blob storage or Azure Data Lake Storage and create external tables in Azure Synapse to that data so that it is still online.
- Consider the possibility of introducing a virtual machine (VM) on Azure with a development version (usually free) of the existing legacy data warehouse DBMS running on this VM. This allows you to quickly move existing data warehouse

schema to the VM and then move it on into Azure Synapse while working entirely on the Azure cloud.

- Define migration order and dependencies.
- Ensure your infrastructure and operations teams are ready for the migration of your data as early as possible into the migration project.
- Identify the differences in DBMS functionality and where proprietary business logic could become a problem. For example, using stored procedures for ELT processing is unlikely to migrate easily and won't contain any metadata lineage since the transformations are buried in code.
- Considering a strategy to migrate data marts first followed by the data warehouse that is the source to the data marts. The reason for this is that it enables incremental migration, it makes it more manageable and it is possible to prioritize migration based on business needs.
- Considering the possibility of using data virtualization to simplify your current data warehouse architecture before you migrate, for example, to replace data marts with virtual data marts so that you can eliminate physical data stores and ETL jobs for data marts without losing any functionality prior to migration. Doing this would reduce the number of data stores to migrate, reduce copies of data, reduce the total cost of ownership and improve agility. This requires switching from physical to virtual data marts before migrating your data warehouse. In many ways, you could consider this a data warehouse modernization step prior to migration.

Next steps

For more information on data warehouse migrations, attend a virtual [cloud data warehouse modernization workshop on Azure](#) from Informatica.

Strategic Migration Assessment and Readiness Tool (SMART)

Article • 02/28/2023

From business planning to training to security and governance, find out what you need to do to prepare for your Microsoft Azure migration with the Strategic Migration Assessment and Readiness Tool (SMART).

The assessment walks you through a series of questions to help assess your readiness in regards to:

- Your current business strategy
- Your partner support system
- Your inventory and assets
- Your business goals
- Your migration plan
- Your organization's technical and migration skills
- Your landing zone environment
- The governance you have in place
- How you will manage assets and services in the cloud

You have the ability to pick an area of focus, or answer all the assessment questions. When you finish, you'll receive a report that identifies areas for improvement and resources that can help you make those improvements. After you make improvements, you can come back and take the assessment again to gage your progress toward migration readiness.

Access the assessment

You can start the assessment by visiting <https://aka.ms/smartzool>.

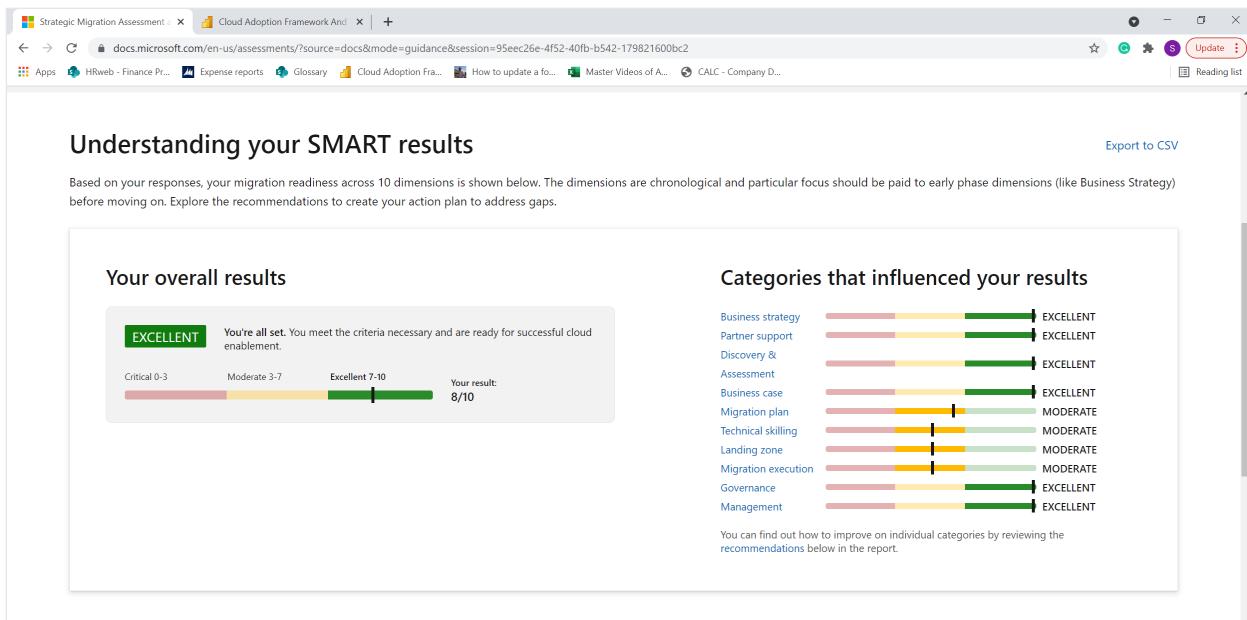
The screenshot shows the Microsoft Docs interface for the Strategic Migration Assessment and Readiness Tool. The page title is "Strategic Migration Assessment and Readiness Tool". It displays a list of 17 questions under four categories: Business strategy, Partner support, Discovery & Assessment, and Business case. A purple callout box at the top right encourages users to sign in to save their progress. The right side of the screen shows a summary of the assessment details: "Strategic Migration Assessment and Readiness Tool - Aug 27, 2021 - 2:00:48 PM". A "Next →" button is visible at the bottom right of the main content area.

The skills necessary to fill the gaps in your organization's digital transformation typically won't be confined to a single role or even a single department. Skills will have relationships and dependencies that can span a single role or multiple roles. Those roles might exist in several departments. For example, a workload owner might require someone in an IT role to create core resources like subscriptions and resource groups.

To ensure quality results, there are two ways we recommend you approach the 17 questions to ensure a smooth experience. The first option is to complete the assessment with a Microsoft account member. The second option is to set aside about 30 minutes with experts in each respective category and then complete the assessment yourself.

Results

After you finish running the SMART tool, a results page is generated. The results page offers tailored recommendations and next steps to improve your score in each category. There's a tool to export the results to a CSV file that allows you to see scores offline.



Recommended next steps

Learn at the Azure migration center

Explore the migration topic with a curated set of resources, how-to articles, videos and tools. Understand how you can do migration yourself or get connected with expert help.

[Azure migration center >](#)

Understand our approach with Cloud Adoption Framework

Microsoft offers proven guidance across every phase of a cloud migration project with considerations for business, technical and organizational aspects.

Join the Azure Migration and Modernization Program

Begin your migration journey with help from Microsoft. We offer expert help from Azure engineering and specialized migration partners, technical skill building and offers to help you reduce you...

There's a matrix that shows your areas of strength and your opportunities for growth based on how you score in each area. You receive a grade across all 10 dimensions that give you a better understanding of what to do next. There's a **Recommended next steps** section and individual recommendations for each category.

To review your answers, you can go to the answer summary. The answer summary can help you understand why you received specific recommendations or what contributed to the individual scores.

Assessment milestones

The milestone feature of the SMART tool allows you to measure your growth and improvement as you take the assessment multiple times. When you use the new milestone feature, you can look at how you improved individually in each area.

The screenshot shows the 'Assessment Overview' page from the Microsoft Cloud Adoption Framework. At the top, there's a header bar with the title 'Strategic Migration Assessment' and a progress bar indicating '17 of 17 questions / 0 skipped'. Below the header, there's a section for 'Milestones' with a table showing two entries. The first entry is 'Milestone 1' with a score of '0 Critical / 0 Moderate / 10 Excellent' saved on April 14, 2021. The second entry is another 'Strategic Migration Assessment and Readiness Tool' entry with a score of '0 Critical / 4 Moderate / 6 Excellent' saved on Feb 26, 2021. A 'Milestone Score History' chart shows a single data point at a score of approximately 8.5, with a date of Feb 26, 2021, 5:14:05 PM. The bottom of the screen shows the Windows taskbar with various pinned apps like File Explorer, Edge, and Mail.

You can compare across all 10 dimensions to see resources that contributed to your evolution.

Next steps

- Strategic Migration Assessment and Readiness Tool (SMART)
- Find out how ready you are to start migrating - the SMART way ↗

Sustainability considerations in cloud adoption planning

Article • 06/03/2024

Planning for sustainability in your cloud journey and business is crucial for several reasons. First, it can help your organization reduce its environmental impact by optimizing resource usage, reducing energy consumption, and lowering carbon emissions. Doing so can demonstrate your commitment to Corporate Social Responsibility (CSR), comply with regulations and industry standards, and improve your brand reputation.

Identify current emissions

Identify current emissions to plan a sustainable cloud adoption journey. Measure over time when moving workloads to/from cloud providers to reduce carbon footprint and achieve compliance with industry standards.

Identify the most carbon-emitting workloads

One key criterion for prioritization of migration workload can be current carbon emissions. From a cloud efficiency and sustainability perspective, the highest emitters should be prioritized to migrate first to the cloud, maximizing the ROI early in the migration journey. Consider using a tool like the [Emissions Savings Estimator](#) which will help you understand the emissions from your current on-premises workloads.

To learn more about available sustainability tools, see [Sustainability tools and resources](#).

Roles and responsibilities

To successfully integrate sustainability into your cloud journey, it's essential to nominate a lead responsible for your sustainability efforts, define clear RACI (Responsibility, Accountability, Consulted, and Informed) for sustainability targets, and build green teams with specific sustainability metrics. The sustainability lead can also engage individual service or application teams to monitor and optimize carbon emissions using [Azure carbon optimization](#). This approach helps businesses to have a sustainability focal point, ensures accountability for meeting targets, and encourages cross-functional collaboration.

Nominate a sustainability lead

Have you defined the person leading the charge for sustainability in your organization?

The role of the Chief Sustainability Officer is evolving. It will likely be pivotal in the future as organizations navigate the complexity of delivering on Environmental, Social, and Governance (ESG) commitments.

It can be the Chief Sustainability Officer if that role exists in your organization. However, it can be anyone ultimately responsible and accountable for sustainability in the cloud. See [making sustainability part of everybody's job](#).

Consider nominating a lead for your sustainability efforts and teams, helping your business have a sustainability focal point. Organizations often have small silos of interested parties, and sustainability is such a task that requires everyone to be participating and work together. Some organizations are beginning their ESG journey, and there might not be a dedicated budget for a Chief Sustainability Officer role. Instead, building a virtual community across the business and dedicating someone to take the lead is a significant first step to consider.

Define a clear RACI

Consider evolving the traditional organization structure by introducing sustainability practices within the company. For example, the Cloud Center of Excellence (CCoE) can encompass the directions for setting up the sustainability practice.

The governance team can define the accountabilities around ensuring the sustainability targets are [being tracked and progressed](#).

Skilling

To drive sustainability awareness, it's essential to upskill employees on green skills and encourage them to participate in related discussions. The sustainability teams can focus on skilling ambitions and should engage the entire organization in relevant initiatives.

Build a sustainability community

Teams need to be aware of new advancements in sustainability constantly. Building a community around cloud efficiency and green software is a good starting point to foster awareness and drive a sustainability culture across your organization. See [how do I start a sustainability community in my organization?](#).

The sustainability community members might be separate from the central organization's sustainability team. However, strategic messaging should flow to the community. Additionally, feedback, initiatives, and suggestions should be fed from the community back into the teams accountable for sustainability. See [sustainability community and knowledge sharing](#) in the Azure Well-Architected Framework.

Learning resources

Sustainability, or cloud efficiency, is a fundamental topic to understand optimizing your cloud infrastructure, applications, and workloads. Knowing how early in your cloud journey you can contribute to becoming more sustainable is also an essential part of your planning process.

- Learn how to design Azure solutions and workloads to be more sustainable. See the [Azure Well-Architected Framework sustainability workload guidance](#).
- See the Microsoft Learn module [The Principles of Sustainable Software Engineering](#).
- You might also benefit from reading the report on [Closing the Sustainability Skills Gap: Helping businesses move from pledges to progress ↗](#).
- Learn more about why educating your teams about sustainability is essential. See [make sustainability part of everybody's job ↗](#).
- For leaders, learn how to integrate sustainability into the organization by aligning climate action with business models. See the LinkedIn training [Navigating Environmental Sustainability: A Guide for Leaders ↗](#).
- Find detailed information about environmental sustainability topics and technologies at the [Microsoft Sustainability Learning Center ↗](#)

Define reporting standards

Create reporting standards to provide consistency and clarity in emissions reporting, which helps in decision-making for sustainability. For instance, use metrics such as Power Usage Effectiveness (PUE) and Greenhouse Gas (GHG) emissions to compare on-premises data centers to cloud services. Also, consider indirect emissions in the supply chain lifecycle.

Reporting standards like the Greenhouse Gas Protocol can help align reporting across organizations. Defining such standards can alleviate resistance to implementing green IT and sustainability practices, particularly in cases where legacy systems need updates.

Planning for reporting standards also helps with [sustainability considerations](#) in your cloud management processes.

Risk management

Manage climate risk effectively by aligning a company's sustainability progress and commitments with the interest of its stakeholder. Considerations include the environmental impact of the company's cloud operations. Understanding this and taking proactive steps to mitigate climate risk can lead to not only protecting themselves from potential adverse impacts but also positioning themselves as leaders in cloud efficiency and sustainability, as well as responsible business practices.

Feedback

Was this page helpful?

 Yes

 No

Cloud adoption plan antipatterns

Article • 03/22/2023

Customers often experience antipatterns while adopting a cloud solution. Either the solution is ineffective or there are unintended consequences. Typical scenarios include:

- Misaligned operating models lead to increased time to market, misunderstandings, and increased pressure on IT departments.
- Companies sometimes choose the wrong service model when they assume that platform as a service (PaaS) decreases costs.
- When an organization's architecture changes, major replacement projects can result. Managing these projects is often complex and cost intensive.

Antipattern: Choose the wrong cloud operating model

A company's strategic priorities and the scope of its portfolio determine its cloud operating model. Models can have different types of accountability, landing zones, and focus. When models don't line up with company goals, problems can result:

- Increased time to market
- Misunderstandings
- Increased pressure on IT departments

Example: Assign too much responsibility to a small team

A corporation introduces an operating model that makes the IT department accountable for everything that runs inside the cloud. The team that's responsible for the cloud contains three people. This setup leads to a slow adoption journey, because:

- The team only approves measures after fully understanding their effect on the business, operations, and security.
- These issues aren't the team's main area of expertise.

Subject matter experts would like to use the cloud service, so business units increase pressure. Shadow IT will probably emerge as business units use company credit cards to create environments for themselves.

Preferred outcome: Compare models and build a readiness plan

Review strategic priorities, portfolio scope, requirements, and constraints. Explore operating model options by [comparing the four most common cloud operations patterns](#) with your current cloud operating model. Identify one or more cloud operating models that suit your organization. Then decide on a model. Because roles change with operating models, [build a skills readiness plan](#) before moving to the cloud.

Antipattern: Choose the wrong service model

Companies sometimes assume that PaaS solutions cost less than infrastructure as a service (IaaS) solutions. This assumption can lead to the wrong choice of service model. Cost-conscious companies often make this mistake when their main reason for moving to the cloud is to lower costs. These companies forget that they also need to change processes when they adopt PaaS, especially when they move certain responsibilities to cloud providers. Switching to PaaS introduces fundamental changes in coordination efforts, engineering practices, and delivery pipelines. Unexpected cost increases and delays can result.

Example: Choose PaaS over IaaS

A publisher launches a program to migrate its datacenters to the cloud. The executives would like to modernize their current application architecture and tooling all at once. Their reasons include:

- Maximizing cost efficiency.
- Developing a more modern application portfolio.

For their adoption strategy, they choose PaaS over IaaS. A year into their cloud adoption journey, they have a slow adoption rate. They've had to change numerous processes, practices, and tools to adopt PaaS to the full extent. The board doesn't see the usual impacts and benefits associated with PaaS. At the same time, IT is slower than ever, while datacenter costs remain the same.

Preferred outcome: Minimize disruption to your business

To reduce coordination efforts, start with IaaS for initial cloud adoption projects. Adopting new processes and practices is more manageable when you move to the cloud later instead of at the outset. Adopt IaaS first, especially in datacenter transformation scenarios. At the same time, launch a cloud skills initiative.

Gradually modernize and adopt PaaS later, after the workload is already in the cloud. The experience that you've gained will help you adopt PaaS faster. You'll need to learn

fewer new skills and processes for modernization. You also won't significantly disrupt your business processes.

Antipattern: Replace architecture

Applications that are based on PaaS and software as a service (SaaS) are relatively easy to maintain. They usually require little effort from management. As a result, many companies redesign old, complex architecture landscapes by replacing them with SaaS and cloud-native concepts. This architecture change usually leads to major replacement projects. It's a complex, cost-intensive task to manage and execute these projects. Changing processes and the operating model also involves other substantial risks.

Example: Choose replacement over modernization

A corporation has a large SAP environment. The IT department would like to replace this landscape, which is causing several performance and stability issues. After IT starts on a replacement project, the due diligence list for replacing the entire environment gets longer every day.

Preferred outcome: Rationalize your digital estate

Before you replace a large or complex application environment, consider incrementally improving your environment by modernizing instead. Relatively small changes to your application environment can have a huge impact on performance and reliability. For instance, changing the hosting platform to Azure can provide stability and quick results. Improved performance and reliability result, at a fraction of the estimated replacement cost.

When deciding on an innovation strategy, explore different modernization options. Evaluate these options in a proof of concept (POC).

Understand your company's [digital estate](#), and evaluate digital assets according to the [five Rs of cloud rationalization](#). Determine which of the *five Rs* works best for modernizing or migrating your assets:

- Rehost
- Refactor
- Rearchitect
- Rebuild
- Replace

Next steps

Learn about the [Ready phase of cloud adoption](#).

Prepare for cloud adoption

Article • 12/14/2023

Before adoption can begin, you create a landing zone to host the workloads that you plan to build in or migrate to the cloud. This section of the framework guides you through environment preparation and landing zone creation.

Landing zone concepts

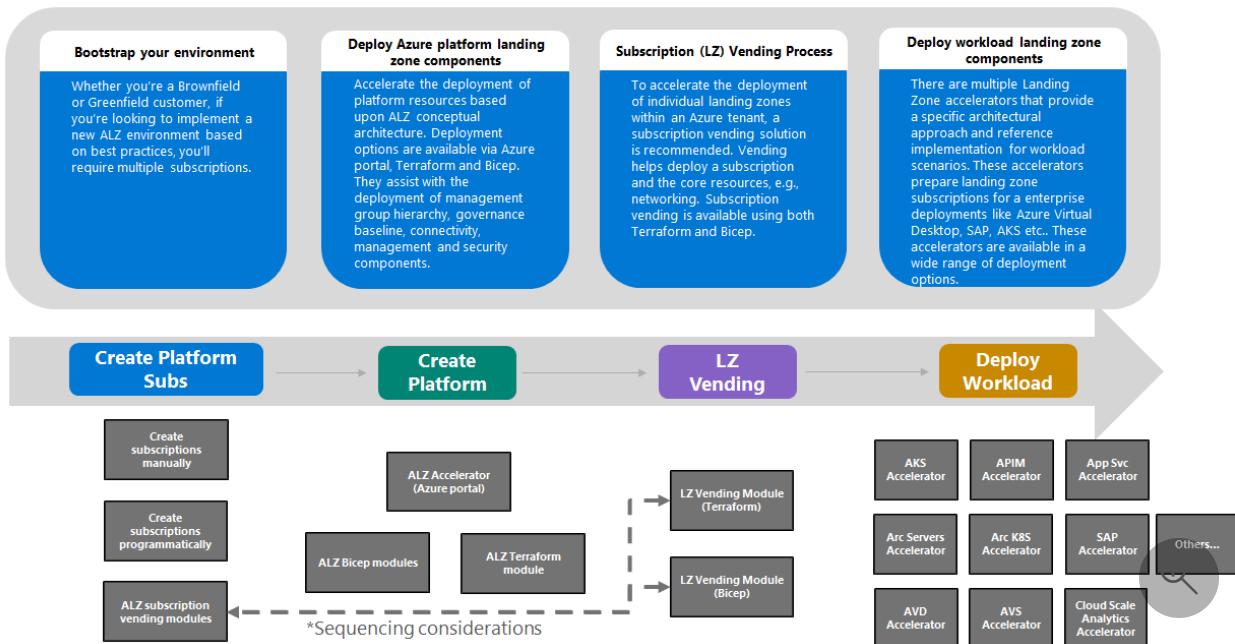
If you understand Azure landing zones, you can skip ahead to the next section. If not, here are some concepts to review before proceeding:

- Abstractly speaking, a ***landing zone*** helps you plan for and design an Azure deployment, by conceptualizing a designated area for placement and integration of resources. There are [two types of landing zones](#):
 - ***platform landing zone***: provides centralized enterprise-scale foundational services for workloads and applications.
 - ***application landing zone***: provides services specific to an application or workload.
- Concretely, a landing zone can be viewed through two lenses:
 - **reference architecture**: a specific design that illustrates resource deployment to one or more Azure subscriptions, which meet the requirements of the landing zone.
 - **reference implementation**: artifacts that deploy Azure resources into the landing zone subscription(s), according to the reference architecture. Many landing zones offer multiple deployment options, but the most common is a ready-made Infrastructure as Code (IaC) template referred to as a ***landing zone accelerator***. Accelerators automate and accelerate the deployment of a reference implementation, using IaC technology such as ARM, Bicep, Terraform, and others.
- A workload deployed to an application landing zone integrates with and is dependent upon services provided by the platform landing zone. These infrastructure services run workloads such as networking, identity access management, policies, and monitoring. This operational foundation enables migration, modernization, and innovation at enterprise-scale in Azure.

In summary, [Azure landing zones](#) provide a destination for cloud workloads, a prescriptive model for managing workload portfolios at scale, and consistency and governance across workload teams.

Landing zone journey

Azure Landing Zone Customer Journey



As you work your way through the Ready guide, consider your progress as a continuous journey that prepares you for landing zone creation. The journey consists of four major phases and related processes:

- Bootstrap your environment
 - [Create subscriptions manually](#)
 - [Create subscriptions programmatically](#)
 - [Subscription vending modules](#)
- Deploy Azure platform landing zone components
 - [Accelerator portal](#)
 - [Bicep modules ↗](#)
 - [Bicep Accelerator ↗](#)
 - [Terraform module ↗](#)
- Subscription landing zone vending process
 - [Vending module \(Terraform\) ↗](#)
 - [Vending module \(Bicep\) ↗](#)
- Deploy workload landing zone components
 - [Cloud adoption scenarios and related accelerators](#)

The phases and processes are covered in more detail as you progress through the Ready guide.

Next steps

Continue with cloud adoption and preparing your Azure environment for landing zones, by reviewing the [Azure setup guide](#).

Azure setup guide overview

Article • 09/18/2023

ⓘ Note

This guide provides a starting point for readiness guidance in the Cloud Adoption Framework. For an interactive experience, view this guide in the Azure portal. Go to the [Azure Quickstart Center](#) in the Azure portal, select the **Projects and guides** tab, then select **Azure Setup Guide** and follow the step-by-step instructions.

Before you start building and deploying solutions by using Azure services, you need to prepare your environment. In this guide, we introduce features that help you organize resources, control costs, and secure and manage your organization. For more information, best practices, and considerations related to preparing your cloud environment, see the [Cloud Adoption Framework's readiness section](#).

You'll learn how to:

- ✓ [Organize resources](#): Set up a management hierarchy to consistently apply access control, policy, and compliance to groups of resources and use tagging to track related resources.
- ✓ [Manage access](#): Use Azure role-based access control to make sure that users only have the permissions they need.
- ✓ [Manage costs and billing](#): Identify your subscription type, understand how billing works, and learn how to control costs.
- ✓ [Plan for governance, security, and compliance](#): Enforce and automate policies and security settings that help you follow applicable legal requirements.
- ✓ [Use monitoring and reporting](#): Get visibility across resources to find and fix problems, optimize performance, and gain insight into customer behavior.
- ✓ [Stay current with Azure](#): Track product updates to enable a proactive approach to change management.

Next steps:

[Organize your resources to simplify how you apply settings](#)

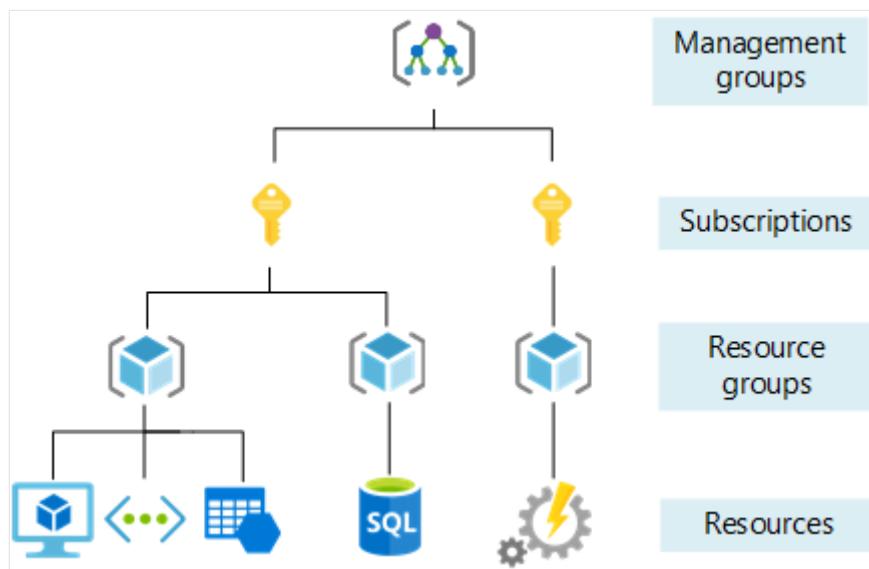
Organize your Azure resources effectively

Article • 05/31/2024

Organize your cloud-based resources to secure, manage, and track costs that are related to your workloads. To organize your resources, define a management group hierarchy, consider and follow a naming convention, and apply resource tagging.

Management levels and hierarchy

Azure provides four levels of management: management groups, subscriptions, resource groups, and resources. The following diagram shows the relationship between these levels.



- **Management groups** help you manage access, policy, and compliance for multiple subscriptions. All subscriptions in a management group automatically inherit the conditions that are applied to the management group.
- **Subscriptions** logically associate user accounts with the resources that they create. Each subscription has limits or quotas on the amount of resources that it can create and use. Organizations can use subscriptions to manage costs and the resources that are created by users, teams, and projects.
- **Resource groups** are logical containers where you can deploy and manage Azure resources like virtual machines, web apps, databases, and storage accounts.
- **Resources** are instances of services that you can create in a resource group, such as virtual machines, storage, and SQL databases.

Note

To minimize the effect of regional outages, we recommend that you place resources in the same region as the resource group. For more information, see [Resource group location alignment](#).

Management settings scope

You can apply management settings, such as policies and role-based access control, at any management level. The level determines how widely the setting is applied. Lower levels inherit settings from higher levels. For example, when you apply a policy to a subscription, that policy applies to all resource groups and resources in that subscription.

Usually, it makes sense to apply critical settings at higher levels and project-specific requirements at lower levels. For example, to make sure that all resources for your organization deploy to certain regions, apply a policy to the subscription that specifies the allowed regions. The allowed locations are automatically enforced when users in your organization add new resource groups and resources.

Managing a few subscriptions independently is easy. However, for a larger number of subscriptions, consider creating a management group hierarchy to simplify management of subscriptions and resources. For more information, see [Organize and manage multiple Azure subscriptions](#).

Work with people in the following roles as you plan your organizational compliance strategy:

- Security and compliance
- IT administration
- Enterprise architecture
- Networking
- Finance
- Procurement

Create a management structure

To create a management group, subscription, or resource group, sign in to the [Azure portal](#).

- To create a *management group* to help you manage multiple subscriptions, go to [Management groups](#) and select **Create**.
- To create a *subscription* to associate users with resources, go to [Subscriptions](#) and select **Add**.

 **Note**

You can also create subscriptions programmatically. For more information, see [Programmatically create Azure subscriptions](#).

- To create a *resource group* to hold resources that share the same permissions and policies:
 1. Go to [Create a resource group](#).
 2. In the **Create a resource group** form:
 - a. For **Subscription**, select the subscription in which to create the resource group.
 - b. For **Resource group**, enter a name for the new resource group.
 - c. For **Region**, select a region in which to locate the resource group.
 3. Select **Review + create**, and after the review passes, select **Create**.

Naming standards

A good naming standard helps to identify resources in the Azure portal, on a billing statement, and in automation scripts. Your naming strategy should include business and operational details in resource names.

- Business details should include the organizational information that's required to identify teams. Use the resource's short name, along with the names of the business owners who are responsible for the resource costs.
- Operational details in resource names should include information that IT teams need. Include details that identify the workload, application, environment, criticality, and other information that's useful for managing resources.

[Naming rules and restrictions](#) vary by the type of resource. For more information and for recommendations that support cloud adoption by enterprises, see [Develop your naming and tagging strategy for Azure resources](#).

 **Note**

- Avoid using special characters, such as hyphen and underscore (- and _), as the first or last characters in a name. Doing so can cause validation rules to fail.
- Names of tags are case-insensitive.

Resource tags

Tags can quickly identify your resources and resource groups. You apply tags to your Azure resources to logically organize them by categories. Tags can include context about the resource's associated workload or application, operational requirements, and ownership information.

Each tag consists of a name and a value. For example, you can apply the name *environment* and the value *production* to all the resources in production.

After you apply tags, you can easily retrieve all the resources in your subscription that have that tag name and value. When you organize resources for billing or management, tags can help you retrieve related resources from different resource groups.

Other common uses for tags include:

- **Metadata and documentation:** Administrators can easily see detail about the resources they're working on by applying a tag like *ProjectOwner*.
- **Automation:** Regularly running scripts can take action based on a tag value like *ShutdownTime* or *DeprovisionDate*.
- **Cost optimization:** You can allocate resources to the teams and resources who are responsible for the costs. In [Microsoft Cost Management](#), you can apply your cost center tag as a filter to report charges based on usage by team or department.

Each resource or resource group can have a maximum of 50 pairs of tag names and values. This limitation is only for tags that directly apply to the resource group or resource.

For more tagging recommendations and examples, see [Develop your naming and tagging strategy for Azure resources](#).

Apply a resource tag

To apply one or more tags to a resource group:

1. In the Azure portal, go to [Resource groups](#) and select the resource group.
2. Select **Assign tags** in the navigation at the top of the page.

3. Enter the name and value for a tag under **Name** and **Value**.
4. Enter more names and values or select **Save**.

Remove a resource tag

To remove one or more tags from a resource group:

1. In the Azure portal, go to [Resource groups](#) and select the ellipses menu for the group, and then select **Edit tags**.
2. Select the trash can icon for each tag that you want to remove.
3. To save your changes, select **Save**.

Next steps

To learn more about management levels and organization, see:

- [Azure fundamentals](#)
- [Create your initial Azure subscriptions](#)
- [Create additional subscriptions to scale your Azure environment](#)
- [Organize and manage multiple Azure subscriptions](#)
- [What are Azure management groups?](#)
- [Resource access management in Azure](#)
- [Azure subscription and service limits, quotas, and constraints](#)

For more information about resource naming and tagging, see:

- [Develop your naming and tagging strategy for Azure resources](#)
- [Use tags to organize your Azure resources and management hierarchy](#)

Feedback

Was this page helpful?

 Yes	 No
---	--

Select Azure regions

Article • 04/01/2024

When you design your strategy to use Microsoft Azure, you can choose from many Azure regions around the world. Region selection is a key part of your overall cloud adoption strategy. Each [Azure region](#) has specific characteristics, so it's essential to choose the best regions for your Azure resources.

Understand Azure region architectures and resilience

Different Azure regions have different characteristics. Two common ways that Azure regions vary involve availability zones and paired regions. Also, some regions are operated by sovereign entities in particular countries. The *region architecture* refers to how a specific region is designed and the overall regional capabilities that it provides.

To learn more about how Azure regions work, see [What are Azure regions and availability zones?](#).

Availability zones

Many Azure regions include availability zones, which are physically separate locations within a region. By using availability zones, you can achieve higher availability and resilience in your deployments. For more information about availability zones, see [Availability zone service support](#) and [Availability zone region support](#).

Paired regions

Some regions are [paired with another region](#), with both regions typically located in the same geopolitical area. Region pairing provides resiliency during catastrophic region failures. Region pairing is mostly used for [geo-redundant storage \(GRS\)](#) and by other Azure services that depend on Azure Storage for replication.

Newer regions aren't paired. Instead, they use availability zones for high availability and resiliency. Later in this article, you learn more about how to use these region types.



Tip

To learn how to design a workload that uses regions and availability zones, see [Recommendations for using availability zones and regions](#).

Sovereign regions

Some regions are dedicated to specific sovereign entities. Although all regions are Azure regions, these sovereign regions are isolated from the rest of Azure. Microsoft doesn't necessarily manage them, and they can be restricted to certain types of customers.

These sovereign regions are [Azure China 21Vianet](#) and [Azure Government - US](#).

Sovereign regions are built to the same standards of resiliency as other Azure regions.

Consider region service availability and capacity

Azure offers two types of regions:

- *Recommended* regions are suitable for most workloads.
- *Alternate* regions aren't optimized for primary workloads. Instead, alternate regions are available only for backup or failover, or only for customers with a company presence within a defined country/region.

When deciding on a region, it's a good idea to select a recommended region if you can, because of the following benefits:

- **Recommended regions typically have higher capacity.** Because of the larger capacity, recommended regions can often support your long-term growth better than alternate regions can.
- **Lower costs.** Many recommended regions provide lower costs for a range of Azure services. By using recommended regions, you might reduce your overall Azure bill.
- **Gain early access to the latest offerings.** For example, AI capabilities and GPU resources are typically available in recommended regions sooner than in other regions.

Microsoft regularly reassesses the regions that we recommend. To take advantage of the benefits of newly recommended regions, consider adopting a [multi-region strategy](#). This strategy helps to ensure you're ready to use more regions for your own workloads.

The services that you can deploy in a region depend on the region's type, among other factors. For more information, see the following resources:

- [Available services by region types and service categories](#)

- [Products available by region](#)

Some regions are reserved for customers who need disaster recovery within their country/region. To request access to these reserved access regions, [create a new support request](#).

Azure is a massively scalable platform, but each region has a maximum capacity. A region's maximum capacity can affect which types of subscriptions can deploy what types of services and under what circumstances. Regional capacity is different from a subscription quota. If you're planning a deployment or migration to Azure, it's a good idea to speak with your local Azure field team or your Azure account manager. Ask for confirmation that you can deploy at the scale that you need.

When you use regions for disaster recovery purposes, consider whether the destination region provides the capacity that you need to support your workloads. For workloads that are based on virtual machines (VMs), consider using [capacity reservations](#) to guarantee the availability of capacity in the regions that you use.

Understand data residency

Around the world, government organizations have begun to establish data sovereignty and data privacy regulations. These types of compliance requirements often require localization in a specific country/region to protect the citizens in that location. In some cases, data that pertains to customers, employees, or partners must be stored on a cloud platform in the same region as the user.

Ensure that you understand your own data residency requirements. Also, verify that the Azure regions that you select are in geographic locations that meet your requirements. For more information, see [Enabling Data Residency and Data Protection in Microsoft Azure Regions](#).

Addressing data residency challenges is a significant motivation for organizations that operate on a global scale to migrate to the cloud. To maintain data sovereignty compliance, some organizations choose to deploy duplicate IT assets to cloud providers in their selected region.

[Microsoft Cloud for Sovereignty](#) is a solution that enables governments to deploy workloads in the Microsoft Cloud while helping meet their specific sovereignty, compliance, security, and policy requirements. Microsoft Cloud for Sovereignty creates software boundaries in the cloud to establish the extra protection that governments require, using hardware-based confidentiality and encryption controls. For more information, see [Microsoft Cloud for Sovereignty capabilities](#).

Consider region proximity

Users or services that need to access your Azure services might reside in various geographies globally. Similarly, your Azure services might need to consume services from external sources that are located in various geographies. Or your services might need to connect to your on-premises systems.

Proximity is an important factor to consider when you select an Azure region. If you use Azure ExpressRoute to connect to your on-premises systems, you can optimize network connectivity and reduce latency by using a region that's close to your on-premises systems. Subsequent connections between Azure regions use the high-speed Microsoft global network.

For more information about latency between Azure regions and other geographic areas, see [Azure network round-trip latency statistics](#).

Operate in multiple geographic regions

It's common for an organization to operate in multiple geographic regions. An organization can get the following benefits by using multiple Azure regions:

- **Run different workloads in different regions.** This reason applies when you want to be close to a specific customer base or business partner. It's also relevant when you want to use Azure services that aren't available in a specific Azure region.
- **Support a geographically dispersed user base.** If you operate in multiple countries, or if your customers use your services from multiple countries, it can make sense to have Azure resources in each location. Alternatively, you can consider using a single region and then using [Azure Front Door](#) to accelerate global traffic to that region.
- **Comply with data sovereignty requirements.** Your organization might be subject to limits on the geographic areas where you can store certain data.
- **Achieve high resiliency,** especially for business-critical workloads. Business-critical workloads require the benefits that availability zones provide, such as high availability and protection from region-wide outages and disasters.
- **Improve network connectivity and performance.** In a hybrid or multicloud scenario, using multiple Azure regions can help improve your network performance. Traffic can enter and exit the high-speed Microsoft backbone network at locations that are close to your on-premises systems, or to another cloud provider's locations. To learn more about multicloud solutions, see [Connectivity to other cloud providers](#).

- **Optimize costs.** Different Azure resource types can have different prices in different regions. When you use tools like the [Pricing calculator](#) and the [Azure service pricing information](#), ensure that you select the correct region to view accurate pricing information. Sometimes you can reduce costs by deploying your development and test environments into a different region. But you need to ensure that the region provides the capabilities and services that you use in your production region.
- **Scale beyond resource quotas.** Some Azure resources have [quotas and limits](#) that restrict the number of instances of a resource that you can create in each region under each subscription. To scale beyond these limits, you might need to use extra subscriptions or multiple regions.
- **Avoid capacity restrictions.** Occasionally, regions have capacity restrictions applied to them. If you use multiple regions, it's probably going to be easier for you to find and use a region that supports the services you want to deploy. If you use a single region and need to expand to a second region just to avoid capacity restrictions, it might take more time for you to prepare and deploy your resources.
- **Reduce complexity compared to multicloud deployments.** The complexity of managing multi-region deployments is typically less than multicloud deployments, and you can often gain similar availability and resiliency benefits. However, the choice between the two approaches depends on your organization's specific goals.

When operating a cloud environment that's spread over multiple geographic regions, consider the following factors:

- **Operational complexity.** When you have multiple resources in different regions, there might be additional operational overhead. You might also have to pay extra costs when you duplicate resources across regions.
- **Data synchronization.** Understand whether you need to synchronize or replicate data between regions. If you do, understand whether to do it asynchronously or synchronously. Configuring a multi-region data storage tier can be complex. You need to consider tradeoffs between resiliency, performance, and cost.
- **Global networking topology.** Azure provides many different networking services. Azure also supports the implementation of various global networking topologies to meet different requirements and provide different tradeoffs. For example, you can expand Azure networking to multiple regions by using [Azure Virtual WAN](#), or you can use a [traditional hub-and-spoke model](#) with some extra effort.
- **User access profiles.** If a single user works with components in multiple regions, understand how to manage their identities and access profiles across regions.
- **Compliance requirements.** Verify that each region satisfies your compliance requirements, including requirements for data sovereignty.

- **Regional resiliency.** Even though using a multi-region architecture helps to increase resiliency, you should also design your solution to be highly available within each region. Use availability zones where you can, and ensure that you consider how to achieve high resiliency within each region.
- **Failover.** When you use multiple regions for resiliency purposes, you can design your solution to use an active-passive approach. This approach requires you to detect regional outages and fail over traffic between regions. It can take time for a failover process to detect an outage and complete traffic routing, which can result in downtime for your services. Some organizations instead choose to deploy in an active-active pattern to avoid relying on failover. The benefits of using an active-active pattern include global load balancing, increased fault tolerance, and network performance boosts. To take advantage of this pattern, your applications must support running simultaneously in multiple regions.

Relocate across regions

Occasionally, you might need to relocate resources or workloads from one Azure region to another. Changes in business requirements, company acquisitions, data residency laws, and other factors are reasons for needing to relocate.

💡 Tip

Relocating resources across regions can be complex. When possible, aim to deploy your resources into the correct region from the start.

Azure provides several tools and various relocation capabilities, but the details vary for each Azure service. Some resource types can be [directly moved across regions](#), and others can be moved by using [Azure Resource Mover](#). Some resource types can't be moved and must be redeployed.

To learn more about relocating across regions, see [Relocate cloud workloads](#).

High availability and disaster recovery across regions

Azure regions are highly available. Azure service-level agreements are applied to the services that run in specific regions. This section provides some considerations that apply if you choose to deploy across multiple regions to increase your resiliency.

Warning

When you design business-critical workloads, always plan for regional failures, and avoid deploying within a single region. You should also practice recovery and mitigation steps. For more information, see [Mission-critical workloads](#).

Understand Azure service resiliency features

Many platform as a service (PaaS) services rely on their own regional resiliency solutions. For example, when you deploy Azure SQL Database and Azure Cosmos DB, you can easily replicate your data to more regions. Other services are deployed to a single region, and you need to manually deploy them to other regions. Also, some Azure services, like Azure DNS and Azure Front Door, are deployed globally and don't have regional dependencies.

For each Azure service that you consider for your cloud adoption process, understand the required failover capabilities and recovery steps.

Plan Azure resource group deployments

For the most reliable scenario and to minimize the effect of regional outages, we recommend that you place resources in the same region as the resource group. For more information, see [Resource group location alignment](#).

If you have resources in different regions within the same resource group, consider moving your resources to a [new resource group or subscription](#).

To [determine if your resource supports moving to another resource group](#), inventory your resources by cross-referencing them. Ensure that you meet the appropriate prerequisites.

Tip

Whenever possible, deploy resource groups in a region that has multiple availability zones. Availability zones help to minimize the risk of regional outages that reduce the availability of your resource and also make management operations unavailable.

In some cases, resources in a resource group span multiple regions. If a whole region is unavailable, all management operations that involve resources within the unavailable

region's resource groups can fail. But resources that are deployed in a different region might remain available even though they can't be managed. In some scenarios, to ensure that resources always remain available, you might place a resource group in multiple regions. This approach has limitations but maintains resource availability during a temporary outage.

Use GRS in paired regions

If you deploy into a region that has an associated paired region, you can use the paired region as part of your multi-region resiliency strategy. Paired regions enable you to [use primary and secondary regions](#).

[Azure Storage supports GRS](#). In Storage GRS, three copies of your data are stored in your primary region, and three more copies are stored in the paired region. You can't change the storage pairing for GRS. Other Azure services that rely on Storage often take advantage of this paired region capability. Your applications and your network must be configured to support paired regions and to use GRS storage appropriately.

Don't attempt to use Storage with GRS replication for your VM backups. Instead, use [Azure Backup](#), [Azure Site Recovery](#), and [Azure managed disks](#) to support resiliency for your infrastructure as a service (IaaS) workloads.

Tip

Multi-region solutions don't have to use Storage GRS. Instead, several other options are available:

- Run your application tier access multiple regions.
- Use a globally distributed database service like [Azure Cosmos DB](#) or [SQL Database](#).
- Use [blob object replication](#).
- Use another multi-region deployment approach.

In these scenarios, when you select a secondary region, consider using a region that isn't the paired region. If a regional failure occurs in your primary region, intense pressure is put on resources in the paired region when resources are migrated and cross-region failover occurs. You can avoid that pressure by recovering to an alternate region, which means you gain speed during your recovery.

Deploy to regions without a pair

Newer Azure regions have no regional pair. They achieve high availability by using availability zones. Such regions follow data residency guidelines that provide the option of storing data in the region.

When you use these regions, you can use locally redundant storage (LRS) or zone-redundant storage (ZRS). Regions without a pair don't support GRS. Services like Backup that have a dependency on Storage might also require that you use ZRS or LRS storage. When possible, it's a good practice to use ZRS to improve your resiliency within your region.

To prepare for the rare event that an entire Azure region is unavailable, you need to plan for cross-region disaster recovery. At a minimum, it's a good practice to deploy your infrastructure by using automation approaches, and to back up your data across regions. If a full-region outage occurs, you can manually redeploy your resources and restore your backups. For some scenarios, you might need to consider other alternatives to reduce your potential recovery time and data loss. For more information, see [Availability zone service support](#), [Availability zone region support](#), and [Azure Resiliency – Business Continuity and Disaster Recovery](#).

Consider your data resiliency needs. Regardless of where your data is located, you can move, copy, or access your data from any location globally.

Some Azure services provide a way for you to store or replicate your data in multiple regions without the regions being paired. For example:

- [Azure Cosmos DB provides global data distribution](#).
- [SQL Database provides active geo-replication to another Azure region](#).
- [Site Recovery supports recovery to any region](#).
- [Azure NetApp Files provides cross-region replication](#).

Next steps

When you migrate existing workloads from an on-premises datacenter to Azure, there are some other region-selection considerations that you should also keep in mind. For more information, see [Select Azure regions for a migration](#).

Feedback

Was this page helpful?

 Yes

 No

Manage access to your Azure environment with Azure role-based access control

Article • 12/01/2022

Managing who can access your Azure resources and subscriptions is an important part of your Azure governance strategy. Assigning group-based access rights and privileges is also a good practice. Dealing with groups instead of individual users simplifies maintenance of access policies, provides consistent access management across teams, and reduces configuration errors. Azure role-based access control (Azure RBAC) is the primary method of managing access in Azure.

Azure RBAC lets you manage access of your resources in Azure. It helps you manage who has access to Azure resources, what they can do with those resources, and what scopes they can access.

When you plan your access control strategy, grant users the least privilege required to get their work done. The following image shows a suggested pattern for assigning Azure RBAC.

	Role			
	Reader	Resource-specific or custom role	Contributor	Owner
Scope				
Management Group	 Management Group			
Subscription	 Subscription	Observers		
Resource Group	 Resource Group		Users managing resources	
Resource	 Resource		Automated Processes	Admins

When you plan your access control methodology, try to work with people in your organization. We recommend that you work with people in security and compliance, IT administration, and enterprise architecture.

The Cloud Adoption Framework offers more guidance on using [Azure role-based access control](#) in your cloud adoption efforts.

Grant resource group access

To grant a user access to a resource group:

1. Go to [Resource groups](#).
2. Select a resource group.
3. Select **Access control (IAM)**.
4. Select **+ Add > Add role assignment**.
5. Select a role, and then assign access to a user, group, or service principal.

Grant subscription access

To grant a user access to a subscription:

1. Go to [Subscriptions](#).
2. Select a subscription.
3. Select **Access control (IAM)**.
4. Select **+ Add > Add role assignment**.
5. Select a role, and then assign access to a user, group, or service principal.

Learn more

To learn more, see:

- [What is Azure role-based access control \(Azure RBAC\)?](#)
- [Cloud Adoption Framework: Use Azure role-based access control](#)

Manage costs and billing for your Azure resources

Article • 05/22/2024

Cost management is the process of effectively planning and controlling costs involved in your business. Cost management tasks are typically performed by finance, management, and application teams. Microsoft Cost Management can help you plan with cost in mind. It can also help you to analyze costs effectively and take action to optimize cloud spending.

For more information about integrating cloud cost management processes throughout your organization, see the Cloud Adoption Framework article on how to [track costs across business units, environments, or projects](#).

Manage your costs with Microsoft Cost Management

Microsoft Cost Management provides a few ways to help you predict and manage costs:

- **Analyze cloud costs** helps you explore and analyze your costs. You can view aggregated cost for your account or view accumulated costs over time.
- **Monitor with budgets** allows you to create a budget and then configure alerts to warn you when you're close to exceeding it.
- **Optimize with recommendations** helps identify idle and underused resources so you can take action to reduce waste.
- **Manage invoices and payments** gives you visibility to your cloud investment.

Predict and manage costs

1. Go to [Cost Management + Billing](#).
2. Select **Cost Management**.
3. Explore the features that help to analyze and optimize cloud costs.

Manage invoices and payment methods

1. Go to [Cost Management + Billing](#).
2. Select **Invoices** or **Payment methods** from the **Billing** section in the left pane.

Billing and subscription support

We offer 24-hour access every day for billing and subscription support to Azure customers. If you need assistance to understand Azure usage, create a support request.

Create a support request

To submit a new support request:

1. Go to [Help + support ↗](#).
2. Select **New support request**.

View a support request

To view your support requests and their status:

1. Go to [Help + support ↗](#).
2. Select **All support requests**.

Learn more

To learn more, see:

- [Microsoft Cost Management documentation](#)
- [Cloud Adoption Framework: Track costs across business units, environments, or projects](#)
- [Cloud Adoption Framework: Cost Management discipline](#)

Feedback

Was this page helpful?

 Yes	 No
---	--

Governance, security, and compliance in Azure

Article • 05/14/2024

You can use tools and services like Azure Policy and Microsoft Defender for Cloud to establish corporate policy and plan your governance strategies. These tools and services enforce and automate your organization's governance decisions. Use the [governance benchmark tool](#) before you start your governance planning to identify potential gaps in your organization's cloud governance approach. For more information about how to develop governance processes, see [Govern methodology](#).

Azure Policy

Azure Policy helps you create, assign, and manage policies. These policies enforce rules on your resources so those resources stay compliant with your corporate standards and service-level agreements. Azure Policy scans your resources to identify resources that aren't compliant with corporate policies. For example, you can have a policy that lets only a specific virtual machine (VM) size to run in your environment. When you implement this policy, Azure Policy evaluates existing VMs in your environment and any new VMs that are deployed. The policy evaluation generates compliance events to use for monitoring and reporting.

Use common policies to:

- Enforce tagging for resources and resource groups.
- Restrict regions for deployed resources.
- Restrict expensive SKUs for specific resources.
- Audit the use of important optional features like Azure-managed disks.

Apply a policy

To apply a policy to a resource group:

1. Go to [Azure Policy](#).
2. Select **Assign a policy**.

Learn more

To learn more, see:

- Azure Policy
- Cloud Adoption Framework: Define corporate policy
- Microsoft Cloud for Sovereignty policy portfolio

Feedback

Was this page helpful?

 Yes

 No

Monitoring and reporting in Azure

Article • 03/21/2023

Azure offers many services that together provide a comprehensive solution for collecting, analyzing, and acting on telemetry from your applications and the Azure resources that support them. These services can also monitor critical on-premises resources to provide a hybrid monitoring environment.

Azure Monitor

Azure Monitor provides a single unified hub for all monitoring and diagnostics data in Azure. You can use it to get visibility across your resources. With Azure Monitor, you can find and fix problems, optimize performance, and understand customer behavior.

- **Data collection:** Azure Monitor collects data from [various data sources](#), including: Application, Container, Guest operating system, Azure resource, Azure subscription, Azure tenant, and Azure resource changes. Additionally, Azure Monitor can collect log data from any REST client using the [Data Collector API](#).
- **Insights:** Availability, performance, usage, and health of your web applications are monitored using [Application Insights](#). Further, you can use the insights features of Azure Monitor to monitor your:
 - [Applications](#)
 - [Containers](#)
 - [Virtual machines](#)
 - [Networks](#)
- **Visualization:** Visualizing your monitoring data will greatly help you get an overview of the current posture of your cloud real estate. Make use of visualizations with built-in or custom charts and tables, workbooks, dashboards, or Power BI.
 - Read more about [best practices for analyzing and visualizing data](#).
- **Response:** An effective monitoring strategy often requires an actionable response to critical events in the collected data. You can automate actions by using the built-in [Alerts](#) or [Autoscale](#) capabilities.

Find more solutions in the Azure Marketplace for monitoring other resource types.

To explore Azure Monitor, go to the [Azure portal](#).

Learn more

To learn more, see [Azure Monitor documentation](#).

Stay current with Azure

Article • 07/10/2024

Cloud platforms like Azure change faster than many organizations are accustomed to. This pace of change means that organizations have to adapt people and processes to a new cadence. If you're responsible for helping your organization keep up with change, you might feel overwhelmed at times. The resources listed in this section can help you stay up to date.

Top resources

The following resources can help you stay current with Azure:

- **Azure Service Health:** [Service Health alerts](#) provide timely notifications about ongoing service issues, planned maintenance, and health advisories. These alerts also includes information about Azure features scheduled for retirement.
- **Azure updates:** Review [Azure updates](#) for announcements about product updates. Brief summaries link to additional details, making the updates easy to follow.
- **Azure blog:** The [Azure blog](#) communicates the most important announcements for the Azure platform. Follow this blog to stay up to date on critical information.
- **Service-specific blogs:** Many individual Azure services publish blogs that you can follow if you rely on those services. Find the ones you're interested in via a web search.

Feedback

Was this page helpful?



Understand cloud operating models

Article • 03/22/2023

Adopting the cloud creates an opportunity to revisit how you operate technological systems. This series of articles clarifies cloud operating models and how they might impact your cloud adoption strategy. However, let us clarify the meaning of *cloud operating model*.

Define your operating model

Before deploying your cloud architecture, try to understand how you want to operate in the cloud. Understanding your strategic direction, among people organization, and governance, risk, and compliance needs helps define your cloud operating model. Your landing zones might provide a variety of options to support your operating model. The next few articles share foundational terms and provide examples of common operating models. These models, which are based on actual customer experiences, might help guide your decision about the right Azure landing zone.

What is an operating model?

Before the existence of cloud technologies, teams established operating models to define how technology would support the business. IT operating models have a number of factors. However, these factors remain consistent: alignment to business strategy, organization of people, change management (or adoption processes), operations management, governance/compliance, and security.

IT operating model factors are essential to long-term technology operations.

Some processes remain relevant when technology operations shift to the cloud. However, the processes might change in some ways. Current operating models focus on physical assets in physical locations, which are funded through capital expenditure cycles. These assets support workloads that the business needs to maintain operations. The mission of most operating models is to prioritize the workloads by investing in the stability of physical assets.

How is a cloud operating model different?

The hardware stack is a never-ending cycle. Physical hardware breaks down and performance degrades. The breakdown of hardware rarely aligns with an organization's

capital expenditure budget and planning cycles. Operating in the cloud changes the way you do hardware refreshes and midnight patches. The cloud shifts your focus upstream to operating systems, applications, and data digital assets. The shift from physical to digital also shifts your technology operating model.

As operating models shift to the cloud, you still need the same people and processes. However, the shift is focused on a higher level of operations. If your people no longer focus on server uptime, then their success metrics will change. When security is no longer protected by the four walls of a datacenter, your threat profile changes. The pace at which you manage changes might also change when procurement no longer blocks innovation.

A *cloud operating model* is the collection of processes and procedures that define how you want to operate technology in the cloud.

Purpose of a cloud operating model

When you remove hardware from the unit of operations, you shift focus to digital assets and the workloads they support. As such, the purpose of the operating models shifts from keeping the lights on to ensuring consistent operations.

The [Microsoft Azure Well-Architected Framework](#) does a great job of decomposing workload considerations into a set of common architectural principles: cost optimization, operational excellence, performance efficiency, reliability, and security.

When moving to a higher level of operations, the common architectural principles help reframe the purpose of the cloud operating model. How do we ensure that all assets and workloads in the portfolio balance these architecture principles? What processes are needed to scale the application of those principles?

Reimagine your operating model

If you update your operating model to remove references to procurement, change, operations, or protection of physical assets, then what's left? For some organizations, their operating model is a clean slate. For most organizations, the constraints developed over time are reduced. In either case, there's an opportunity to think about how you want to operate in the cloud.

To help you imagine your future state operating model, these articles discuss the following subjects:

- [Define your cloud operating model](#)

- Compare common cloud operating models
- Implement your operating model with Azure landing zones

Next steps

Learn how the Cloud Adoption Framework helps you define your operating model.

[Compare common cloud operating models](#)

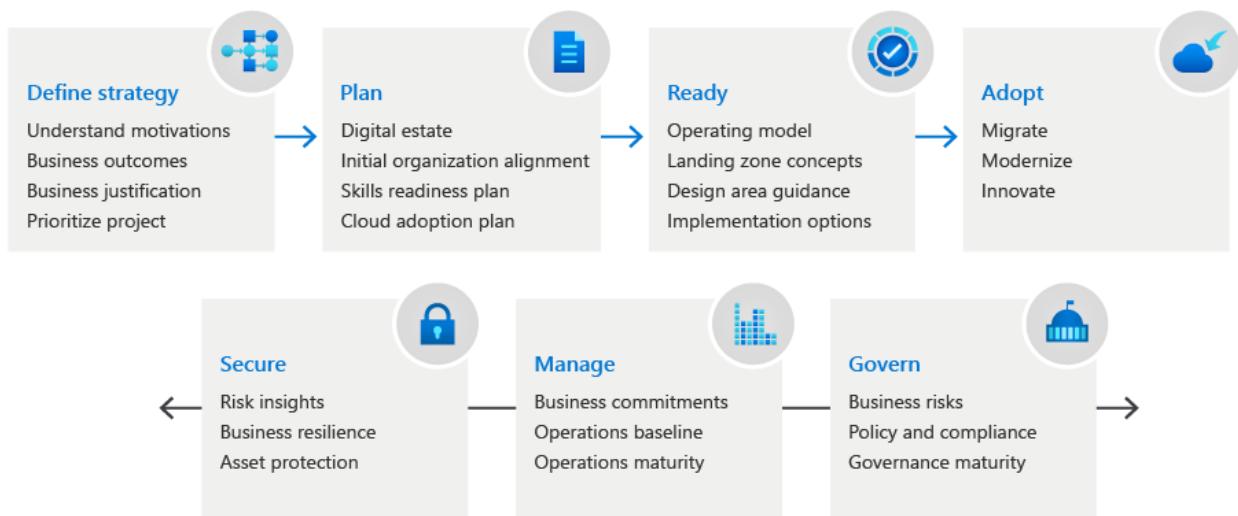
Define your cloud operating model

Article • 10/24/2023

Cloud operating models are complex, and it's easy to fall into a series of circular references when defining them. The Cloud Adoption Framework provides a series of complimentary and incremental methodologies that break many decisions into smaller exercises and help you avoid circular references as you define your organization's cloud operating model.

Cloud Adoption Framework alignment

To help you define the cloud operating model for your business, the Cloud Adoption Framework breaks down each aspect of the operating model into methodologies. Each methodology and its actionable exercises are designed to help you define your future state operations.



Support for developing your operating model

The following incremental methodologies are designed to help you develop your operating Cloud Adoption Framework model.

- **Manage:** Align ongoing processes for operational technology management.
- **Govern:** Maintain alignment with governance and compliance requirements and ensure consistency across your adoption efforts.
- **Secure:** Align your business to the security disciplines and strengthen your security posture.
- **Organize:** Outline which functions your business needs and define organizational methods for your business goals and people.

Collective operating model output

Your environment should represent the way your business operates. As you define your operating model, ensure your environmental readiness aligns with your operations, governance, security, and organizational requirements.

- **Ready:** Use deployment guidance and reference implementations from Azure landing zones to help build your environmental configuration.

ⓘ Note

The Ready methodology provides two implementation options for Azure landing zones:

- **Start small and expand:** Build your cloud platform as you define each aspect of your operating model.
- **Enterprise-scale:** Build an enterprise-ready architecture based on defined operating-model decisions.

Dependencies and inputs for operating model decisions

Consider business strategy and collective cloud adoption plans as you define your operating model.

- **Strategy:** Capture your business strategy and map it to your cloud adoption strategy efforts.
- **Plan:** Establish backlogs and align ongoing changes using Agile-based change management.

Next steps

Before you engage any of the methodologies described in this article, use the following article to compare common cloud operating models. Identify the model that most closely meets your requirements. This closest-match cloud operating model provides an actionable starting point and exercises to move you toward your desired cloud platform operating model.

[Compare common cloud operating models](#)

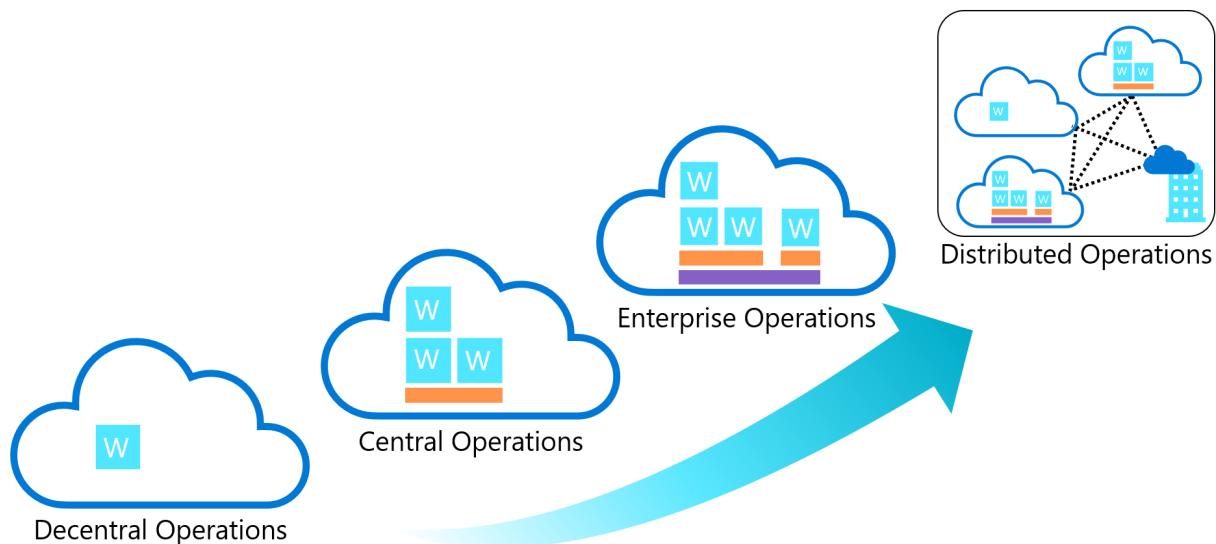
Review and compare common cloud operating models

Article • 03/22/2023

Operating models are unique and specific to the business they support, based on their current requirements and constraints. But operating models aren't *snowflakes*. There are several common patterns in customer operating models; this article outlines the four most common patterns.

Operating model comparison

The following image maps common operating models based on the range of complexity, from least complex (decentralized) to most complex (global operations). The following tables compare the same operating models based on the relative value of a few other attributes.



Priorities or scope

A cloud operating model is primarily driven by two factors:

- Strategic priorities or motivations
- Scope of the portfolio to be managed

Decentralized operations (ops)	Centralized operations (ops)	Enterprise operations (ops)	Distributed operations (ops)
--------------------------------	------------------------------	-----------------------------	------------------------------

	Decentralized operations (ops)	Centralized operations (ops)	Enterprise operations (ops)	Distributed operations (ops)
Strategic priorities or motivations	Innovation	Control	Democratization	Integration
Portfolio scope	Workload	Landing zone	Cloud platform	Full portfolio
Workload environment	High complexity	Low complexity	Medium complexity	Medium or variable complexity
Landing zone	N/A	High complexity	Medium to low complexity	Low complexity
Foundation utilities	N/A	N/A or low support	Centralized and more support	Most support
Cloud foundation	N/A	N/A	Hybrid, provider specific, or regional foundations	Distributed and synchronized

- Strategic priorities or [motivations](#): Each operating model delivers the typical [strategic motivations for cloud adoption](#). But some operating models simplify specific motivations.
- [Portfolio scope](#): The portfolio scope identifies the largest scope that a specific operating model design supports. For example, centralized operations are designed for a few landing zones. But the operating model decision might create operational risks for an organization. Operational risks result when trying to manage a large complex portfolio. These portfolios might require many landing zones or variable complexity in landing zone design.

ⓘ Important

Adopting the cloud often triggers reflection on the current operating model and might lead to a shift from one common operating model to another. But cloud adoption isn't the only trigger. Business priorities and the scope of cloud adoption can change how the portfolio needs to be supported. Also, there could be other shifts in the most-appropriately aligned operating model. When the board, or other executive teams, develop 5 to 10 year business plans, those plans often include a requirement (explicit or implied) to adjust the operating model. Operating models are a good reference for guiding decisions. These models might change or need to be customized to meet your requirements and constraints.

Accountability alignment

Many teams and individuals are responsible for supporting different functions. But each common operating model assigns final accountability for decision outcomes to one team or individual. This approach affects how the operating model is funded and what level of support is provided for each function.

	Decentralized ops	Centralized ops	Enterprise ops	Distributed ops
Business alignment	Workload team	Central cloud strategy	CCoE	Variable - form a broad cloud strategy team?
Cloud operations	Workload team	Central IT	CCoE	Based on portfolio analysis - see Business alignment and Business commitments
Cloud governance	Workload team	Central IT	CCoE	Multiple layers of governance
Cloud security	Workload team	Security operations center (SOC)	CCoE + SOC	Mixed - see Define a security strategy
Cloud automation and DevOps	Workload team	Central IT or N/A	CCoE	Based on portfolio analysis - see Business alignment and Business commitments

Accelerate operating model implementation in Azure

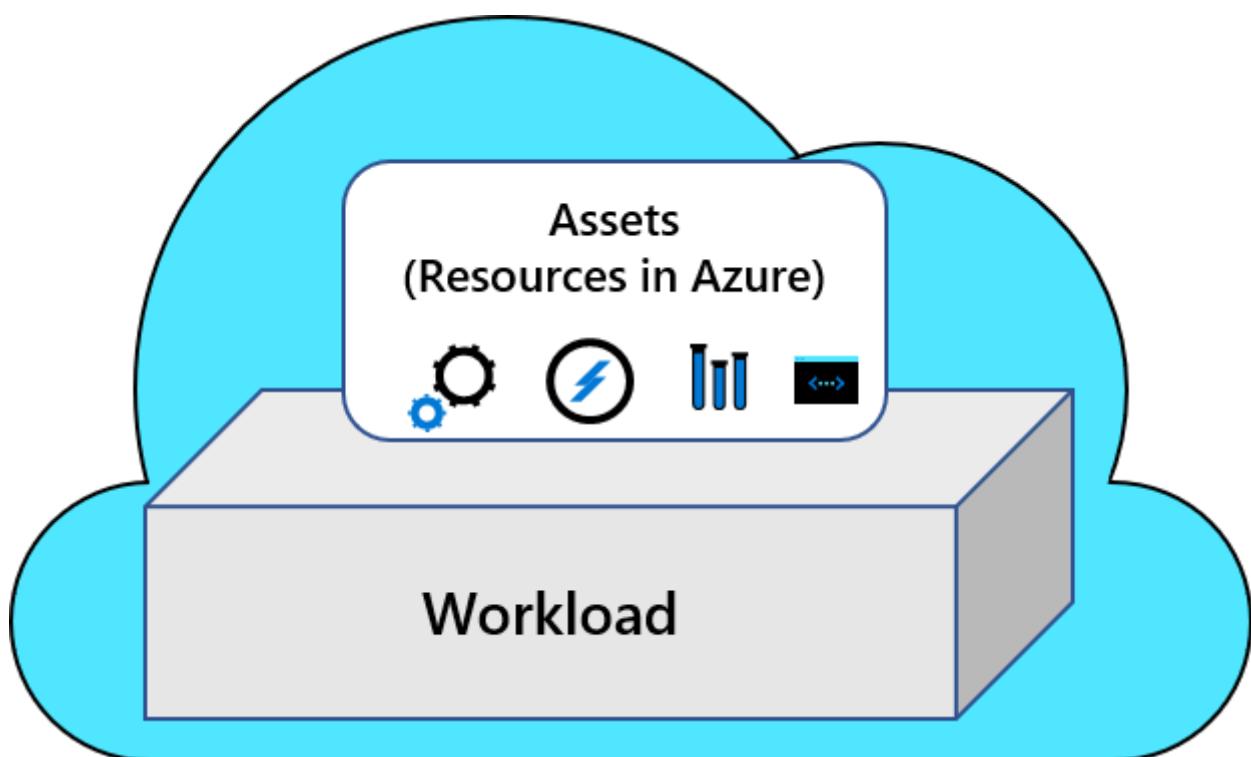
As discussed in [Define your operating model](#), each methodology of the Cloud Adoption Framework provides a structured path for developing your operating model. These methodologies might help you overcome blockers that stem from gaps in adopting the cloud operating model.

The following table outlines ways to accelerate your operating model implementation.

	Decentralized ops	Centralized ops	Enterprise ops	Distributed ops
Starting point	Azure Well-Architected Framework (WAF)	Azure landing zones: start-small options	Azure landing zones: CAF enterprise-scale	Business alignment

	Decentralized ops	Centralized ops	Enterprise ops	Distributed ops
Iterations	A focus on workloads lets the team iterate within WAF.	The start-small option requires more iteration on each methodology but can be done as cloud adoption efforts mature.	As illustrated by the reference implementations, future iterations typically focus on minor configuration additions.	Review the Azure landing zone implementation options to start with the option that best meets your operations baseline. Follow the iteration path defined in that option's design principles.

Decentralized operations



Operations are always complex. If you limit the scope of your operations to one workload or a small collection of workloads, you control the complexity. Decentralized operations are the least complex of the common operating models. In this form of operations, all workloads operate independently by dedicated workload teams.

- **Priorities:** Your team measures innovation over centralized control or standardization across multiple workloads.
- **Distinct advantage:** Maximize speed of innovation by placing workload and business teams in full control of design, build, and operations.
- **Distinct disadvantage:** Reduction in cross-workload standardization, economies of scale through shared services, and consistent governance centralized compliance efforts.

- **Risk:** This approach introduces risk when managing a portfolio of workloads. Workload teams might have specialized teams dedicated to central IT functions. This operating model is viewed as a high risk option by some organizations, especially companies that are required to follow third-party compliance requirements.
- **Guidance:** Decentralized operations are limited to workload-level decisions. Microsoft Azure Well-Architected Framework supports the decisions made within that scope. The processes and guidance within the Cloud Adoption Framework might add overhead that isn't required by decentralized operations.

Advantages of decentralized operations

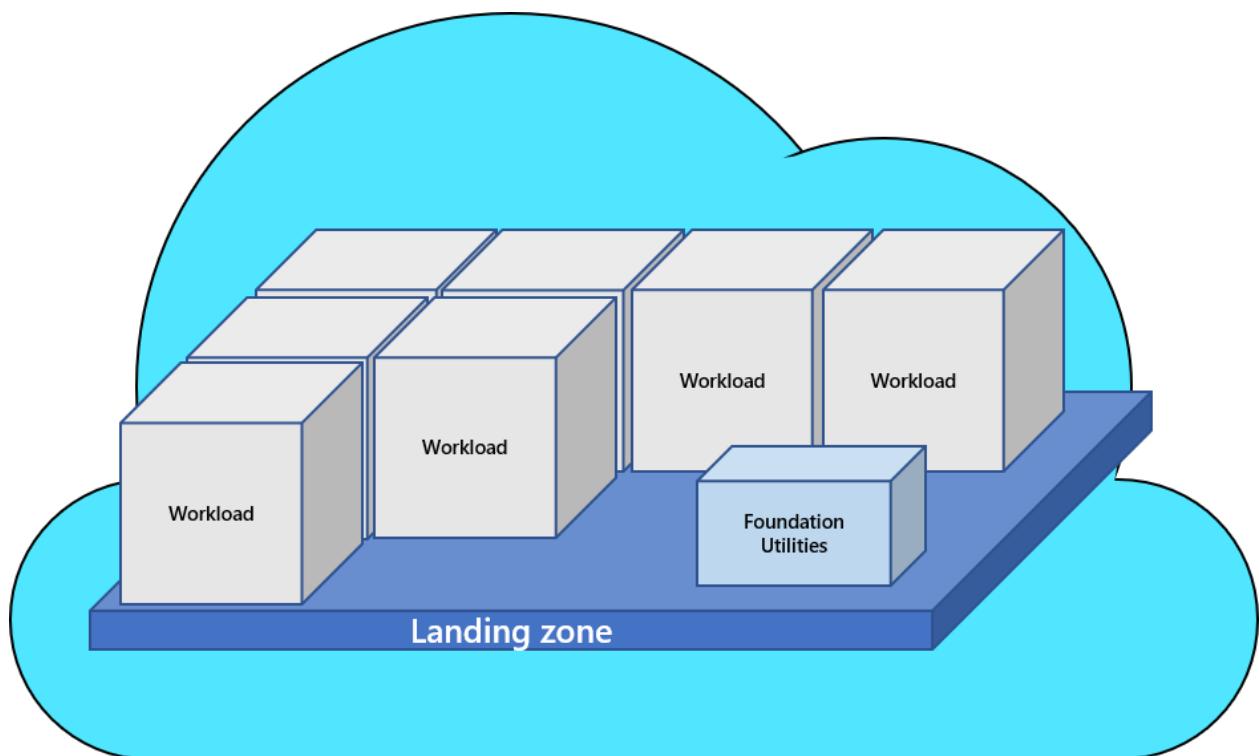
- **Cost management:** Cost of operations is easily mapped to a single business unit. Workload-specific operations support greater workload optimization.
- **Responsibilities:** Typically, this form of operations is highly dependent on automation to minimize overhead. Responsibilities tend to focus on DevOps and pipelines for release management. This type of operations supports faster deployments and shorter feedback cycles during development.
- **Standardization:** Use a source code and deployment pipeline to standardize the environment from release to release.
- **Operations support:** Decisions that affect operations are only concerned with the needs of that workload and simplifying operations decisions. Members of the DevOps community say that operations support is the purest form of operations because of the tighter operational scope.
- **Expertise:** DevOps and development teams are most empowered by this approach and experience the least resistance to driving market change.
- **Landing zone design:** No specific operational advantage.
- **Foundational utilities:** No specific operational advantage.
- **Separation of duties:** No specific operational advantage.

Disadvantages of decentralized operations

- **Cost management:** Enterprise costs are harder to calculate. Lack of centralized governance teams makes it harder to implement uniform cost controls or optimization. At scale, this model can be costly, because each workload might have duplication in deployed assets and staffing assignments.
- **Responsibilities:** Lack of centralized support means that the workload team is entirely responsible for governance, security, operations, and change management. The lack of support is problematic when those tasks haven't been automated in code review and release pipelines.

- **Standardization:** Standardization across a portfolio of workloads is variable and inconsistent.
- **Operations support:** Scale efficiencies are often missed while creating best practices across multiple workloads.
- **Expertise:** Team members have a greater responsibility to make wise and ethical decisions about governance, security, operations, and change management within the application design and configuration. Consult the Microsoft Azure Well-Architected Review and Azure Well-Architected Framework frequently to improve the required expertise.
- **Landing zone design:** Landing zones aren't workload-specific and aren't considered in this approach.
- **Foundational utilities:** Few (if any) foundational services are shared across workloads, reducing scale efficiencies.
- **Separation of duties:** Higher requirements for DevOps and development teams increase usage of elevated privileges from those teams. If you require separation of duties, you might need to heavily invest in DevOps maturity to operate with this approach.

Centralized operations



Stable state environments might not require focus on the architecture or distinct operational requirements of the individual workloads. Central operations tend to be the norm for technology environments that consist primarily of stable-state workloads. Examples of stable-state operations include things like commercial-off-the-shelf (COTS) applications, or well-established custom applications that have a slow release cadence. If

a rate of change is driven by regular updates and patches, the centralization of operations might be an effective way to manage your portfolio.

- **Priorities:** Priorities are the central control over innovation, and measure the existing operational processes over the cultural shift to modern cloud operations.
- **Distinct advantage:** Centralization introduces economies of scale, best-of-breed controls, and standardized operations, and works best with the cloud environment. These environments need specific configurations to integrate cloud operations into existing operations and processes. Centralization is most advantageous with a portfolio of a few hundred workloads with modest architectural complexity and compliance requirements.
- **Distinct disadvantage:** Scaling to meet the demands of a large portfolio of workloads can place significant strain on centralized teams that make operational decisions for production workloads. If technical assets expect to scale beyond 1,000 VMs, applications, or data sources, you might consider an enterprise model if it's within 18-24 months.
- **Risk:** This approach limits centralization to a smaller number of subscriptions (often one production subscription). Significant risk is involved when refactoring later in your cloud journey, and might interfere with your adoption plans. To avoid rework, try focusing on segmentation, environment boundaries, identity tooling, and other foundational elements.
- **Guidance:** Azure landing zone implementation options that are aligned to the "start small and expand" development velocity creates a sound starting point. You can use these options to accelerate adoption efforts. But to be successful, establish clear policies to guide early adoption efforts within acceptable risk tolerances. Governing and Managing methodologies helps create processes to mature operations in parallel. Following these steps serve as stage gates that must be completed before allowing increased risk as operations mature.

Advantages of centralized operations

- **Cost management:** Centralizing shared services across many workloads creates economies of scale and eliminates duplicated tasks. Central teams can quickly implement cost reductions through enterprise-wide sizing and scale optimizations.
- **Responsibilities:** Centralized expertise and standardization might lead to higher stability, better operational performance, and minimal change-related outages. This approach reduces broad skilling pressures on the workload focused teams.
- **Standardization:** In general, standardization and cost of operations is lowest with a centralized model because there are fewer duplicated systems or tasks.
- **Operations support:** Reducing complexity and centralizing operations makes it easier for smaller IT teams to support operations.

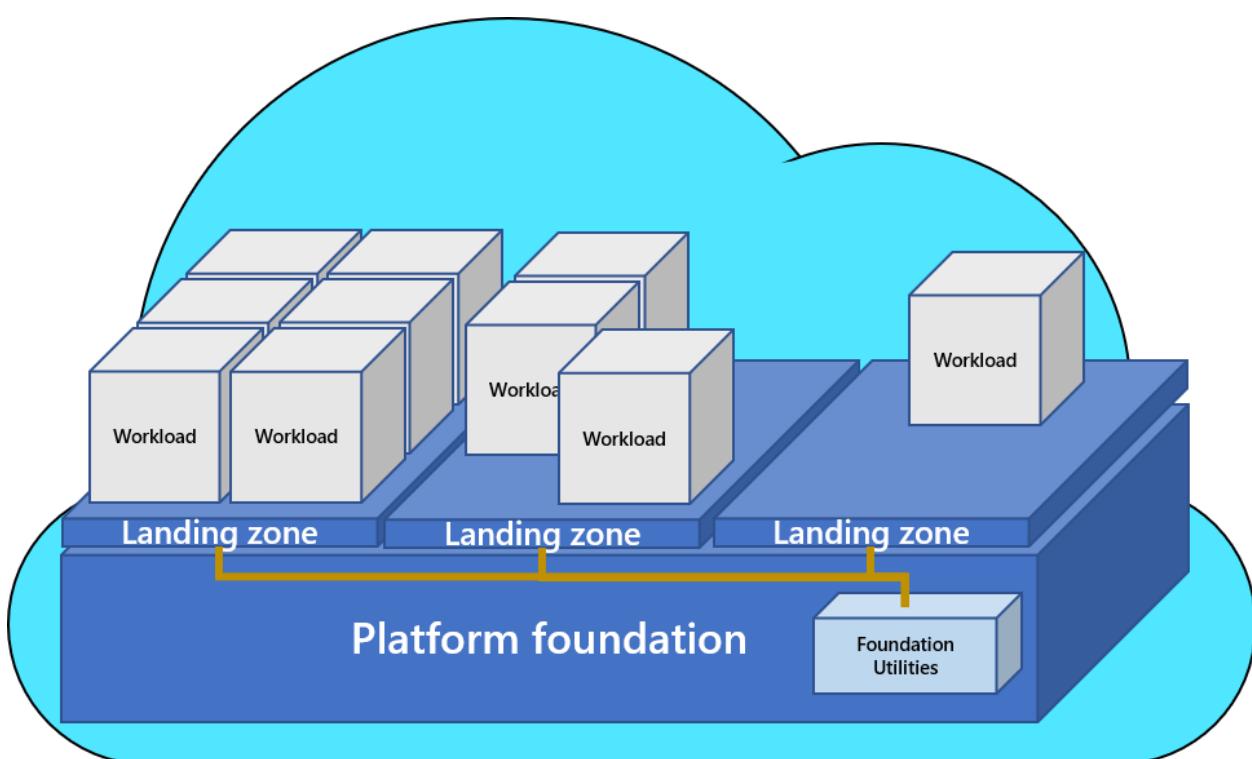
- **Expertise:** Centralizing supporting teams lets experts in security, risk, governance, and operations drive business-critical decisions.
- **Landing zone design:** Central IT reduces complexity by minimizing the number of landing zones and subscriptions. Landing zone designs tend to mimic the preceding datacenter designs, which reduce transition time. As adoption progresses, shared resources might be moved into a separate subscription or platform foundation.
- **Foundational utilities:** You carry existing datacenter designs into the cloud results in foundational, shared services that mimic on-premises tools and operations. When on-premises operations are your primary operating model, it might be an advantage, but beware of some disadvantages. On-premises operations reduce transition time, capitalizes on economies of scale, and supports consistent operational processes between on-premises and cloud hosted workloads. This approach can reduce short-term complexity and effort and let smaller teams support cloud operations with reduced learning curves.
- **Separation of duties:** Separation of duties is clear in central operations. Central IT maintains control of the production environments and reduces the need for any elevated permissions from other teams. This approach reduces breaches by limiting the number of accounts with elevated privileges.

Disadvantages of centralized operations

- **Cost management:** Central teams don't always understand workload architectures to produce impactful optimizations at the workload level. This lack of understanding limits the amount of cost savings that comes from well-tuned workload operations. Not fully understanding workload architecture can affect centralized cost optimizations, which affect performance, scale and other pillars of a well-architected workload. Before you apply enterprise-wide cost changes to high profile workloads, your central IT team should understand and complete the Microsoft Azure Well-Architected Review.
- **Responsibilities:** Centralizing production support and access places high operational burden on a few people and greater pressure on each individual. The pressures placed on these individuals cause the need to perform deeper reviews of the deployed workloads, which validate adherence to detailed security governance and compliance requirements.
- **Standardization:** Central IT approaches make it difficult to scale standardization without a linear scaling of central IT staff.
- **Operations support:** The greatest disadvantages of this approach are associated with significant scale and shifts that measures innovation.

- **Expertise:** Developer and DevOps experts are at risk of being under-valued or too constrained in this type of environment.
- **Landing zone design:** Datacenter designs are based on the constraints of preceding approaches, which aren't always relevant to the cloud. Following this approach reduces the opportunities to rethink environment segmentation and empower opportunities for innovation. Lack of landing zone segmentation increases the potential effect of breach, complexity of governance and compliance adherence, and might create blockers to adoption in the cloud journey. See the risks section above.
- **Foundational utilities:** During digital transformation, cloud might become the primary operating model. Central tools, which are built for on-premises operations, reduce opportunities to modernize operations and increase operational efficiencies. Choosing not to modernize operations early in the adoption process is also an option. Modernizing might be achieved by creating a platform foundation subscription in the cloud adoption journey. That effort can be complex, costly, and time-consuming without advanced planning.
- **Separation of duties:** Central operations generally follow one of two paths and both might hinder innovation.
 - **Option 1:** Teams outside of central IT are granted limited access to development environments that mimic production. This option hinders experimentation.
 - **Option 2:** Teams develops and test in non-supported environments. This option hinders deployment processes and slows post-deployment integration testing.

Enterprise operations



Enterprise operations are the suggested target state for all cloud operations. Enterprise operations balance the need for control and innovation by simplifying decisions and responsibilities. Central IT is replaced by a more facilitative cloud center of excellence or CCoE team, which supports workload teams. The CCoE team, holds workload teams accountable for decisions, as opposed to controlling or limiting their actions. Workload teams are granted more power and more responsibility to drive innovation, within well-defined guardrails.

- **Priorities:** Priorities are the democratization of technical decisions. Democratization of technical decisions shifts responsibilities previously held by central IT to workload teams. To deliver this shift in priorities, decisions become less dependent on human-run review processes. This approach supports automated review, governance, and enforcement, using cloud-native tools.
- **Distinct advantage:** Segmentation of environments and separation of duties allow for balance between control and innovation. Centralized operations maintain workloads that require increased compliance and stable state operations, or represent greater security risks. Conversely, this approach supports reducing centralized control of workloads and environments that require greater innovation. Larger portfolios might struggle with the balance between control and innovation. This flexibility makes it easier to scale thousands of workloads with reductions in operational pains.
- **Distinct disadvantage:** What worked on-premises might not work well in enterprise cloud operations. This approach to operations requires changes on many fronts. Cultural shifts in control and responsibility are often the biggest challenge. Operational shifts that follow the cultural shifts take time and committed efforts to implement, mature, and stabilize. Architectural shifts might be required in stable workloads, while tooling shifts are required to empower and support the cultural, operational, and architectural shifts. These shifts might require commitments to a primary cloud provider. Adoption efforts made before these changes can require significant rework that goes beyond typical refactoring efforts.
- **Risk:** This approach requires executive commitment to the change strategy. It also requires commitment from the technical teams to overcome learning curves and deliver the required change. Long-term cooperation between business, CCoE and central IT, and workload teams is required to see long-term benefits.
- **Guidance:** Azure landing zone options are defined as *enterprise-scale*. These options provide reference implementations to demonstrate how technical changes deliver using cloud-native tooling in Azure. The enterprise-scale approach guides teams through the operational and cultural shifts required to take full advantage of those implementations. That same approach might tailor the reference architecture to configure the environment to meet your adoption strategy and compliance constraints. When you implement enterprise-scale, the Govern and Manage

methodologies can help define processes. These processes can expand your compliance and operations capabilities to meet your operational needs.

Advantages of enterprise operations

- **Cost management:** Central teams act on cross-portfolio optimizations and hold individual workload teams accountable for deeper workload optimization. Workload-focused teams are empowered to make decisions and provided clarity when those decisions have a negative cost effect. Central and workload teams share accountability for cost decisions at the right level.
- **Responsibilities:** Central teams use cloud-native tools to define, enforce, and automate guardrails. Workload team efforts are accelerated through CCoE automation and practices. Workload teams are empowered to drive innovation and make decisions within those guardrails.
- **Standardization:** Centralized guardrails, and foundational services, create consistency across all environments.
- **Operations support:** Workloads that require centralized operations support are segmented to environments with stable-state controls. Segmentation and separation of duties empower workload teams to take accountability for operational support in their own dedicated environments. Automated cloud native tools ensure a minimum operations baseline for all environments with centralized operational support.
- **Expertise:** Centralizing core services such as security, risk, governance, and operations ensures proper central expertise. Clear processes and guardrails educate and empowers workload teams to make more detailed decisions. These decisions expand the effect of the centralized experts without needing to scale staff linearly with technology scale.
- **Landing zone design:** Landing zone design replicates the needs of the portfolio, creating clear security, governance, and accountability boundaries. These boundaries are required to operate workloads in the cloud. Segmentation practices are unlikely to resemble the constraints created by preceding datacenter designs. In enterprise operations, landing-zone design is less complex, allowing for faster scale and reduced barriers to self-service demand.
- **Foundational utilities:** Foundational utilities are hosted in separate centrally controlled subscriptions, known as the platform foundation. Central tools are then piped into each landing zone as utility services. Separating foundational utilities from the landing zones maximizes consistency and economy of scale. These utilities also create clear distinctions between centrally managed responsibilities and workload level responsibilities.

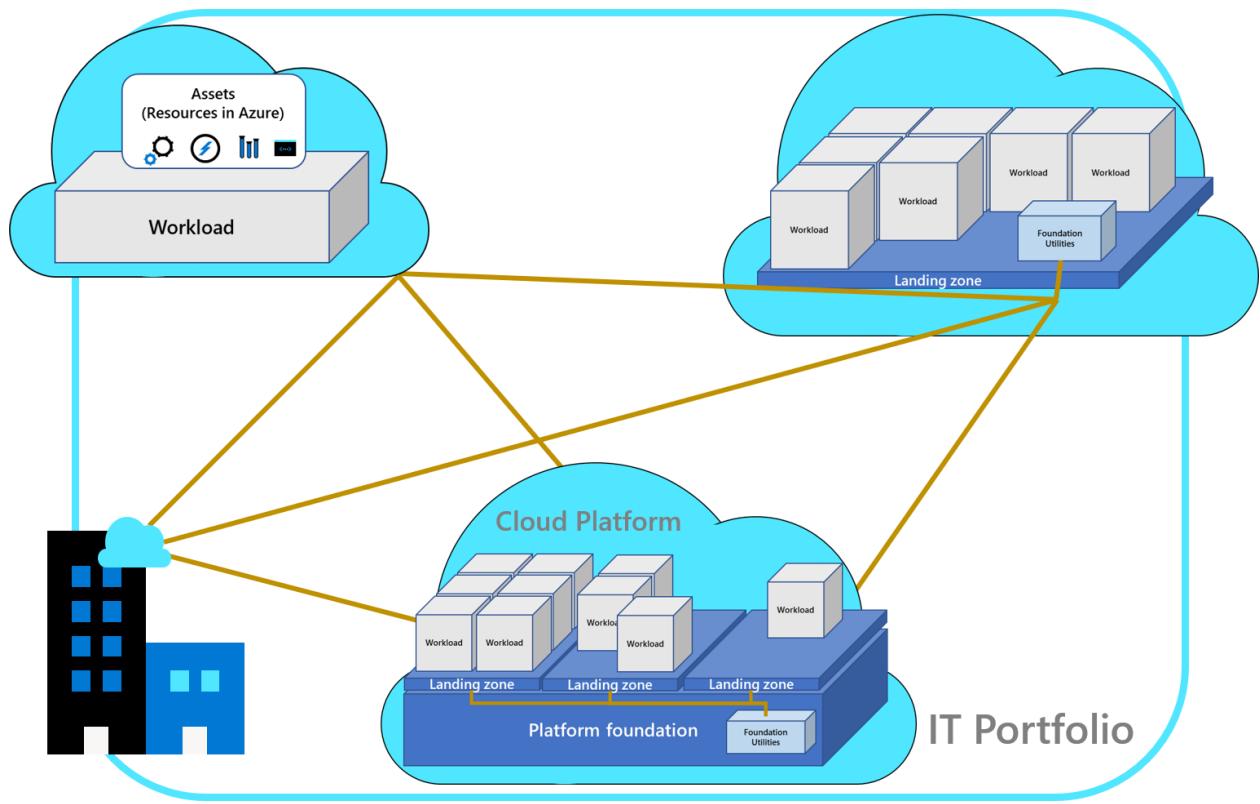
- **Separation of duties:** Clear separation of duties between foundational utilities and landing zones is one of the biggest advantages in the operations approach. Cloud-native tools and processes support access and proper balance of control between centralized teams and workload teams. This approach is based on the requirements of individual landing zones and workloads hosted in landing zone segments.

Disadvantages of enterprise operations

- **Cost management:** Central teams are more dependent on workload teams to make production changes within landing zones. This shift creates a risk for potential budget overruns and slower right-sizing of actual spend. Cost control processes, clear budgets, automated controls, and regular reviews must be in place early to avoid cost surprises.
- **Responsibilities:** Enterprise operations require greater cultural and operational requirements. These requirements ensure clarity in responsibilities and accountability between central and workload teams.
- Traditional change management processes, or change advisory boards (cabs), might not maintain the pace and balance required in this operating model. Those processes are reflected in automating processes and procedures that safely scale cloud adoption.
- Lack of commitment to change materializes first in negotiation and alignment of responsibilities. Inability to align on shifts in responsibility is an indication that central IT operating models might be required during short-term cloud adoption efforts.
- **Standardization:** Lack of investment in centralized guardrails, or automation, create risks to standardization, which is more difficult to overcome through manual review processes. Operational dependencies between workloads in landing zones and shared services creates greater risks. These risks extend from standardization during upgrade cycles or future versions of foundational utilities. During platform foundation revisions, improved, or even automated testing, is required of all supported landing zones and the workloads they host.
- **Operations support:** The operations baseline provided through automation and centralized operations might be sufficient for low affect or low criticality workloads. But workload teams, or other forms of dedicated operations, might be required for complex or high criticality workloads. If so, it might create a shift in operations budgets, requiring business units to give operating expenses to those forms of advanced operations. If central IT is required to maintain sole accountability for the cost of operations, enterprise operations might be difficult to implement.

- **Expertise:** Central IT team members might be required to develop expertise in automating central controls previously delivered via manual processes. Also, these teams might develop proficiency for infrastructure-as-code approaches to defining the environment, and understand branching, merging, and deployment pipelines. At minimum, a platform automation team might need decision-making skills to understand decisions made by cloud center of excellence or central operation teams. Workload teams might be required to develop more knowledge related to the controls and processes that govern their decisions.
- **Landing zone design:** Landing zone design is dependent on foundational utilities. Workload teams should understand what's in the design and what's forbidden to include. This understanding might help avoid duplication of efforts, errors, or conflicts. To create flexibility, you can factor in exception processes to your landing zone designs.
- **Foundational utilities:** Centralizing foundational utilities takes time. These utilities eventually consider options and develop solutions that might scale to meet various adoption plans. Delays in early adoption efforts are possible. Delays might be offset in the long term due to accelerations and blocker avoidance later in the process.
- **Separation of duties:** Ensuring clear separation of duties requires mature identity management processes. There might be more maintenance associated with the proper alignment of users, groups, and onboarding and off-boarding activities. You might need to adopt new processes to accommodate just-in-time access via elevated privileges.

Distributed operations



The existing operating model might be too engrained for the entire organization to shift to a new operating model. For others, global operations and various compliance requirements might prevent specific business units from making a change. In this case, it might require a distributing operations approach. This approach is by far the most complex, as it requires integrating one or more of the previously mentioned operating models.

While heavily discouraged, this operations approach might be required for some organizations. The approach mainly relates to organizations that have a loose collection of disparate business units, a diverse base of customer segments, or regional operations.

- **Priorities:** Integrate multiple existing operating models.
- Transitional state with a focus on moving the entire organization to one of the previously mentioned operating models.
- Longer term operational approach when the organization is too large or too complex to align to a single operating model.
- **Distinct advantage:** Integrate common operating model elements from each business unit. This approach creates a vehicle to group operating units into a hierarchy that helps them mature operations using consistent repeatable processes.
- **Distinct disadvantage:** Consistency and standardization across multiple operating models is difficult to maintain for extended periods. This operational approach requires deep awareness of the portfolio and how various segments of the technology portfolio operate.

- **Risk:** Lack of commitment to a primary operating model could lead to confusion across teams. Use this operating model when there's no way to align to a single operating model.
- **Guidance:** Start with a thorough review of the portfolio, which uses the approach outlined in the [business alignment](#) articles. Try to group the portfolio by the state operating model (decentralized, centralized, or enterprise).
- Develop a management group hierarchy that reflects the operating model groupings. This arrangement includes other organizational patterns for region, business unit, or other criteria that map the workload clusters from least common to most common buckets.
- Evaluate the alignment of workloads to operating models to find the most relevant operating model cluster to start with. Follow the guidance that's mapped to the operating model for all workloads under the node and management group hierarchy.
- Use Govern and Manage methodologies to find common corporate policies, including required operational management practices at various points of the hierarchy. Apply common Azure policies to automate the shared corporate policies.
- As you test Azure policies with various deployments, attempt to move them higher in the management group hierarchy. The policies can be applied to many workloads, which might find commonalities and distinct operation needs.
- Over time, this approach might help you define a model that scales across various operating models. This approach might also unify teams through a set of common policies and procedures.

Advantages and disadvantages of this approach are purposefully blank. After you complete the business alignment of your portfolio, see the predominant operating model section above for clarity on advantages and disadvantages.

Next steps

Learn the terminology associated with operating models. The terminology helps you understand how an operating model fits into the bigger theme of corporate planning.

[Operating model terminology](#)

Learn how a landing zone provides the basic building block of any cloud adoption environment.

[What is a landing zone?](#)

Operating model terminology

Article • 03/22/2023

The term *operating model* has many definitions. This intro article establishes terminology associated with operating models. To understand an operating model as it relates to the cloud, we first have to understand how an operating model fits into the bigger theme of corporate planning.

Terms

Business model: Business models tend to define corporate value (*what* the business does to provide value) and mission/vision statements (*why* the business has chosen to add value in that way). At a minimum, business models should be able to represent the *what* and *why* in the form of financial projections. There are many different schools of thought regarding how far a business model goes beyond these basic leadership principles. However, to create a sound operating model, the business models should include high-level statements to establish directional goals. It's even more effective if those goals can be represented in metrics or KPIs to track progress.

Customer experience: All good business models ground the *why* side of a business's strategy in the experience of their customers. This process could involve a customer acquiring a product or service. It could include interactions between a company and its business customers. Another example could center around the long-term management of a customer's financial or health needs, as opposed to a single transaction or process. Regardless of the type of experience, the majority of successful companies realize that they exist to operate and improve the experiences that drive their *why* statements.

Digital transformation: Digital transformation has become an industry buzzword. However, it's a vital component in the fulfillment of modern business models. Since the advent of the smartphone and other portable computing form factors, customer experiences have become increasingly digital. This shift is painfully obvious in some industries like DVD rentals, print media, automotive, or retail. In each case, digitized experiences have had a significant impact on the customer experience. In some cases, physical media have been entirely replaced with digital media, upsetting the entire industry vertical. In others, digital experiences are seen as a standard augmentation of the experience. To deliver business value (*what* statements), the customer experience (*why* statements) must factor in the impact of digital experiences on the customers' experiences. This process is digital transformation. Digital transformation is seldom the entire *why* statement in a business strategy, but it's an important aspect.

Operating model: If the business model represents the *what* and *why*, then an operating model represents the *how* and *who* for operationalizing the business strategy. The operating model defines the ways in which people work together to accomplish the large goals outlined in the business strategy. Operating models are often described as the people, process, and technology behind the business strategy. In the article on the Cloud Adoption Framework operating model, this concept is explained in detail.

Cloud adoption: As stated above, digital transformation is an important aspect of the customer experience and the business model. Likewise, cloud adoption is an important aspect of any operating model. Cloud adoption is a strong enabler to deliver the right technologies and processes required to successfully deliver on the modern operating model.

Cloud adoption is *what we do* to realize the business value. The operating model represents *who we are* and how we function on a daily basis while cloud adoption is being delivered.

Take action

[Use the operating model](#) provided by the Cloud Adoption Framework to develop operational maturity.

Next steps

Continue to the next section of the Cloud Adoption Framework. Learn how a landing zone provides the basic building block of any cloud adoption environment.

[Use the operating model](#)

What is an Azure landing zone?

Article • 10/07/2024

An Azure landing zone is an environment that follows key design principles across eight design areas. These design principles accommodate all application portfolios and enable application migration, modernization, and innovation at scale. An Azure landing zone uses subscriptions to isolate and scale application resources and platform resources. Subscriptions for application resources are called application landing zones, and subscriptions for platform resources are called platform landing zones.

Azure landing zone architecture

An Azure landing zone architecture is scalable and modular to meet various deployment needs. The repeatable infrastructure allows you to apply configurations and controls to every subscription consistently. Modules make it easy to deploy and modify specific Azure landing zone architecture components as your requirements evolve.

The Azure landing zone conceptual architecture (see *figure 1*) represents an opinionated target architecture for your Azure landing zone. You should use this conceptual architecture as a starting point and [tailor the architecture to meet your needs](#).

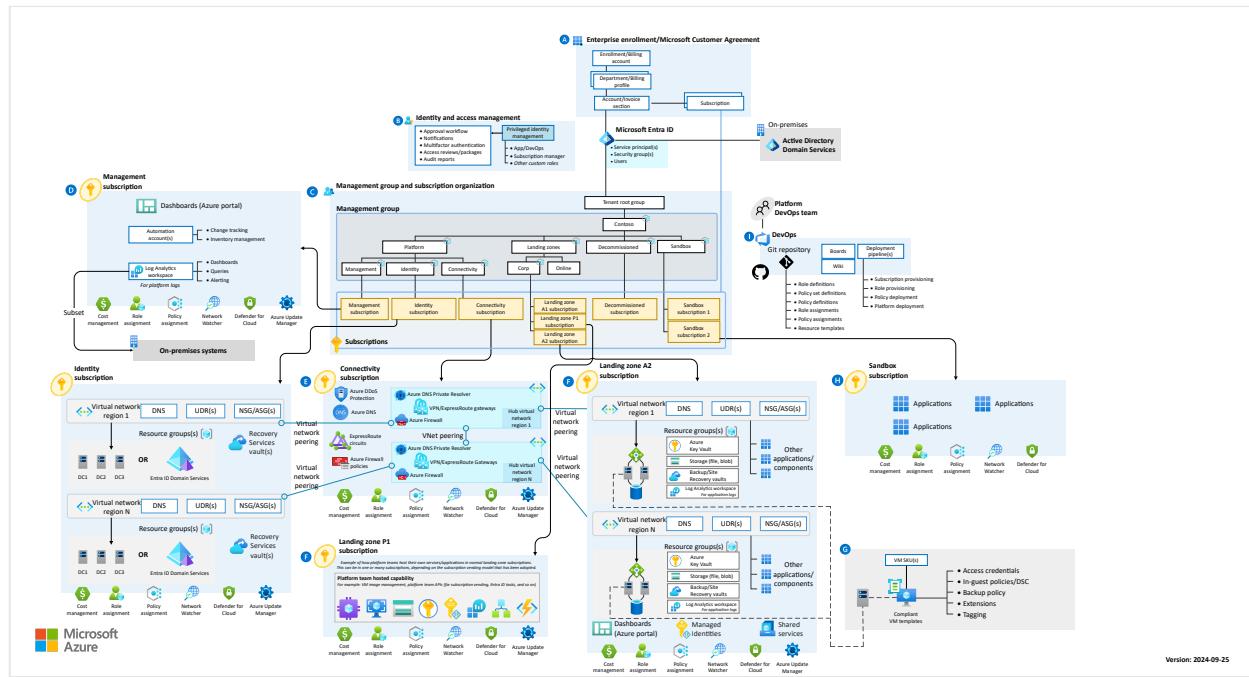


Figure 1: Azure landing zone conceptual architecture. Download a [Visio file](#) of this architecture.

Design areas: The conceptual architecture illustrates the relationships between its eight design areas. These design areas are Azure billing and Microsoft Entra tenant, identity and access management, management group and subscription organization, network

topology and connectivity, security, management, governance, and platform automation and DevOps. For more information on the design areas, see [the Azure Landing Zone environment design areas](#).

Resource organization: The conceptual architecture shows a sample management group hierarchy. It organizes subscriptions (yellow boxes) by management group. The subscriptions under the "Platform" management group represent the platform landing zones. The subscriptions under the "Landing zone" management group represent the application landing zones. The conceptual architecture shows five subscriptions in detail. You can see the resources in each subscription and the policies applied.

Platform landing zones vs. application landing zones

An Azure landing zone consists of platform landing zones and application landing zones. It's worth explaining the function of both in more detail.

Platform landing zone: A platform landing zone is a subscription that provides shared services (identity, connectivity, management) to applications in application landing zones. Consolidating these shared services often improves operational efficiency. One or more central teams manage the platform landing zones. In the conceptual architecture (*see figure 1*), the "Identity subscription," "Management subscription," and "Connectivity subscription" represent three different platform landing zones. The conceptual architecture shows these three platform landing zones in detail. It depicts representative resources and policies applied to each platform landing zone.

Application landing zone: An application landing zone is a subscription for hosting an application. You pre-provision application landing zones through code and use management groups to assign policy controls to them. In the conceptual architecture (*see figure 1*), the "Landing zone A1 subscription" and "Landing zone A2 subscription" represent two different application landing zones. The conceptual architecture shows only the "Landing zone A2 subscription" in detail. It depicts representative resources and policies applied to the application landing zone.

There are three main approaches to managing application landing zones. You should use one of the following management approaches depending on your needs:

- Central team approach
- Application team approach
- Shared team approach

Application landing zone management approach	Description
Central team management	A central IT team fully operates the landing zone. The team applies controls and platform tools to the platform and application landing zones.
Application team management	A platform administration team delegates the entire application landing zone to an application team. The application team manages and supports the environment. The management group policies ensure that the platform team still governs the application landing zone. You can add other policies at the subscription scope and use alternative tooling for deploying, securing, or monitoring application landing zones.
Shared management	With technology platforms such as AKS or AVS, a central IT team manages the underlying service. The application teams are responsible for the applications running on top of the technology platforms. You need to use different controls or access permissions for this model. These controls and permissions differ from the ones you use to manage application landing zones centrally.

Azure landing zone accelerators

Accelerators are infrastructure-as-code implementations that help you deploy an Azure landing zone correctly. We have a platform landing zone accelerator and several application landing zone accelerators you can deploy.

Platform landing zone accelerator

There's a ready-made deployment experience called the [Azure landing zone portal accelerator](#). The Azure landing zone portal accelerator deploys the conceptual architecture (*see figure 1*) and applies predetermined configurations to key components such as management groups and policies. It suits organizations whose conceptual architecture aligns with the planned operating model and resource structure.

You should use the Azure landing zone portal accelerator if you plan to manage your environment with the Azure portal. If you want to use Bicep or Terraform, see the [Bicep and Terraform deployment options](#). Deploying the Azure landing zone portal accelerator requires permissions to create resources at the tenant (/) scope. Follow the guidance in [Tenant deployments with ARM templates: Required access](#) to grant these permissions.



Deploy to Azure



Application landing zone accelerators

[Application landing zone accelerators](#) help you deploy application landing zones. Use the list of available application landing zone accelerators in the [Azure Architecture Center](#) and deploy the accelerator that matches your scenario.

<https://www.microsoft.com/en-us/videoplayer/embed/RE4xdvm?postJs||Msg=true>

Video explaining application landing zones and their implementation principles

Next steps

An Azure landing zone is an environment that adheres to crucial design principles across eight design areas. You should familiarize yourself with these design principles to tailor them to your needs.

[Design principles](#)

Feedback

Was this page helpful?

 Yes

 No

Azure landing zone design principles

Article • 09/27/2024

The Azure landing zone conceptual architecture universally applies to any Azure landing zone process or implementation. At the foundation of the architecture, a set of core design principles serve as a compass for subsequent design decisions across critical technical domains.

The principles are intentionally aspirational, to help you strive for an optimum design of the target architecture. If you choose to deploy an implementation that's an Azure landing zone accelerator, or any version of the enterprise-scale landing zone code base, build on the architecture by applying the design principles this article describes.

Use these principles in your implementation as a useful guide to realize the benefits of cloud technologies. This cloud-oriented or *cloud native* approach represents ways of working and technical options for your organization that legacy technology approaches don't typically offer.

Familiarize yourself with these principles to better understand their impact and the tradeoffs associated with deviation.

Impact of design deviations

There might be valid reasons to deviate from the design principles. For example, organizational requirements might dictate specific outcomes or approaches for designing an Azure environment. In such cases, it's important to understand the impact the deviation has on the design and future operations. Carefully consider the tradeoffs each principle outlines.

As a general rule, be prepared to balance requirements and functionality. Your journey to a conceptual architecture evolves over time as requirements change and you learn from your implementation. For example, using preview services and depending on service roadmaps can remove technical blockers during adoption.

Subscription democratization

Use subscriptions as units of management, and scale to accelerate application migrations and new application development. Align subscriptions with business needs and priorities to support business areas and portfolio owners. Provide subscriptions to

business units to support the design, development, and testing of new workloads and the migration of existing workloads.

To help the organization operate effectively at scale, support a subscription with a suitable [Management Group hierarchy](#). This hierarchy allows efficient subscription management and organization.

💡 Tip

For more information about subscription democratization, see the recent YouTube video [Azure Landing Zones - How many subscriptions should I use in Azure?](#)

Impact of deviation

- **Decentralized vs. centralized operations.** One way to implement this principle transitions operations to business units and workload teams. This reassignment lets workload owners have more control and autonomy over their workloads, within the guardrails of the platform foundation. Organizations that require [central operations](#) might not want to delegate control of production environments to workload teams or business units. These organizations might need to modify their [resource organization](#) design to deviate from this principle.
- **Operating model misalignment.** Azure landing zone conceptual architecture design assumes a specific management group and subscription hierarchy for all operations management subscriptions. This hierarchy might not align with your [operating model](#). As your organization grows and evolves, your operating model might change. Moving resources into separate subscriptions can lead to complicated technical migrations. Review the [Align](#) guidance before you commit to an approach.

Policy-driven governance

Use Azure Policy to provide guardrails and ensure that the applications you deploy comply with your organization's platform. Azure Policy provides application owners with independence and a secure, unhindered path to the cloud.

For more information, review [Adopt policy-driven guardrails](#).

Impact of deviation

Increased operational and management overhead. If you don't use policies to create guardrails within your environment, you increase the operational and management overhead of maintaining compliance. Azure Policy helps you restrict and automate your desired compliance state within your environment.

Single control and management plane

Avoid dependency on abstraction layers such as customer-developed portals or tooling. It's best to have a consistent experience for both central operations and workload operations. Azure provides a unified and consistent control plane that applies across all Azure resources and provisioning channels. The control plane is subject to role-based access and policy-driven controls. You can use this Azure control plane to establish a standardized set of policies and controls that govern your entire enterprise estate.

Impact of deviation

Increased integration complexity. A multivendor approach to control and management planes might introduce integration and feature support complexity. Replacing individual components to achieve a "best of breed" design or multivendor operations tooling has limitations, and could cause unintended errors due to inherent dependencies.

If you're bringing an existing tooling investment to operations, security, or governance, review the Azure services and any dependencies.

Application-centric service model

Focus on application-centric migrations and development, rather than pure infrastructure lift-and-shift migrations such as moving virtual machines. Design choices shouldn't differentiate among old and new applications, infrastructure as a service (IaaS) applications, or platform as a service (PaaS) applications.

Regardless of the service model, strive to provide a secure environment for all applications you deploy on the Azure platform.

Impact of deviation

- **Increased governance policy complexity.** If you segment workloads differently from the management group hierarchy [implementation options](#), you increase complexity in governance policies and access control structures that govern your

environment. Examples include deviation from the organizational hierarchy structure or grouping by Azure service.

- **Increased operational overhead.** This tradeoff introduces the risk of unintentional policy duplication and exceptions, which add to operational and management overheads.
- **Dev/Test/Production** is another common approach that organizations consider. For more information, see [How do we handle "dev/test/production" workload landing zones in Azure landing zone architecture](#).

Alignment with Azure-native design and roadmaps

Use Azure-native platform services and capabilities whenever possible. This approach should align with Azure platform roadmaps to ensure that new capabilities are available within your environments. Azure platform roadmaps should help inform the migration strategy and the Azure landing zone conceptual trajectory.

Impact of deviation

Increased integration complexity. Introducing third-party solutions into your Azure environment can create a dependency on those solutions to provide feature support and integration with Azure first-party services.

Sometimes bringing existing third-party solution investments into an environment is inescapable. Consider this principle and its tradeoffs carefully to align with your requirements.

Next steps

Organizations might be at different stages of their cloud journeys when they review this guidance. Therefore, the required actions and recommendations to progress toward the preceding outcomes might vary. To understand the best next actions for your cloud adoption stage, review the journey to the target architecture.

Journey to the target architecture

To choose the right Azure landing zone implementation option, understand the Azure landing zone design areas.

[Review design areas](#)

Feedback

Was this page helpful?

 Yes

 No

Journey toward the target architecture

Article • 11/28/2024

Adopting cloud technologies is a journey. Business priorities and the need to bring new technologies online to unlock capabilities or features influence the speed at which an organization deploys and scales out a cloud environment.

Over time, an organization iterates and matures the deployed technologies, processes, and skilling needed to progress toward that destination. This iteration doesn't mean you arrive at the destination immediately, however. The journey takes time. The time varies depending on the size of the organization, the current technical footprint, and the skilling maturity within technical teams.

On-ramps

Consider an analogy of a trip along a freeway. There might be multiple *on-ramps* you can use to join the freeway, but the *destination* is the same.



Start



Align



Enhance

These on-ramps represent where your organization is today in your cloud adoption plans. They also represent the specific guidance that you need to continue to develop your cloud environment.

[] [Expand table](#)

On-ramp	Description	Further guidance
Start	Your organization is at the beginning of the cloud adoption journey, also referred to as <i>greenfield</i> , and you want to implement a new cloud environment based on best practices and proven	<ul style="list-style-type: none">- What is an Azure landing zone?- Azure landing zone design areas

On-ramp	Description	Further guidance
	<p>architectural patterns.</p> <p>Start with the Azure landing zone conceptual architecture to understand the recommended end state.</p> <p>Explore each of the design areas. Understand the key themes in each area that form the considerations and decisions that you need to design and implement a landing zone that best fits your requirements.</p>	
Align	<p>Your organization has an existing environment that needs modification to align with the Azure landing zone target architecture and best practices, also referred to as <i>brownfield</i>.</p> <p>See the transition from brownfield guidance to understand the decision points and technical approach to refactor environments and align with guidance in the ready methodology.</p>	<ul style="list-style-type: none"> - Refactor a landing zone - Transition an existing Azure environment to the Azure landing zone conceptual architecture - Scenario: Transition a single subscription with no management groups to the Azure landing zone conceptual architecture - Scenario: Transition management groups to the Azure landing zone conceptual architecture - Scenario: Transition a regional organization environment to the Azure landing zone conceptual architecture - Scenario: Transition an environment by duplicating a landing zone management group
Enhance	<p>Your environment is already in line with best practices, but your organization wants to add more controls or features.</p> <p>Explore articles that describe considerations to enhance key ongoing processes for cloud</p>	<ul style="list-style-type: none"> - Improve landing zone operations - Improve landing zone governance - Improve landing zone security

On-ramp	Description	Further guidance
	environments, such as management, governance, and security.	

Next steps

For considerations about implementing an Azure landing zone, explore the design areas in an Azure landing zone. Consider specific technical subjects for each landing zone.

[Azure landing zone design areas](#)

Feedback

Was this page helpful?

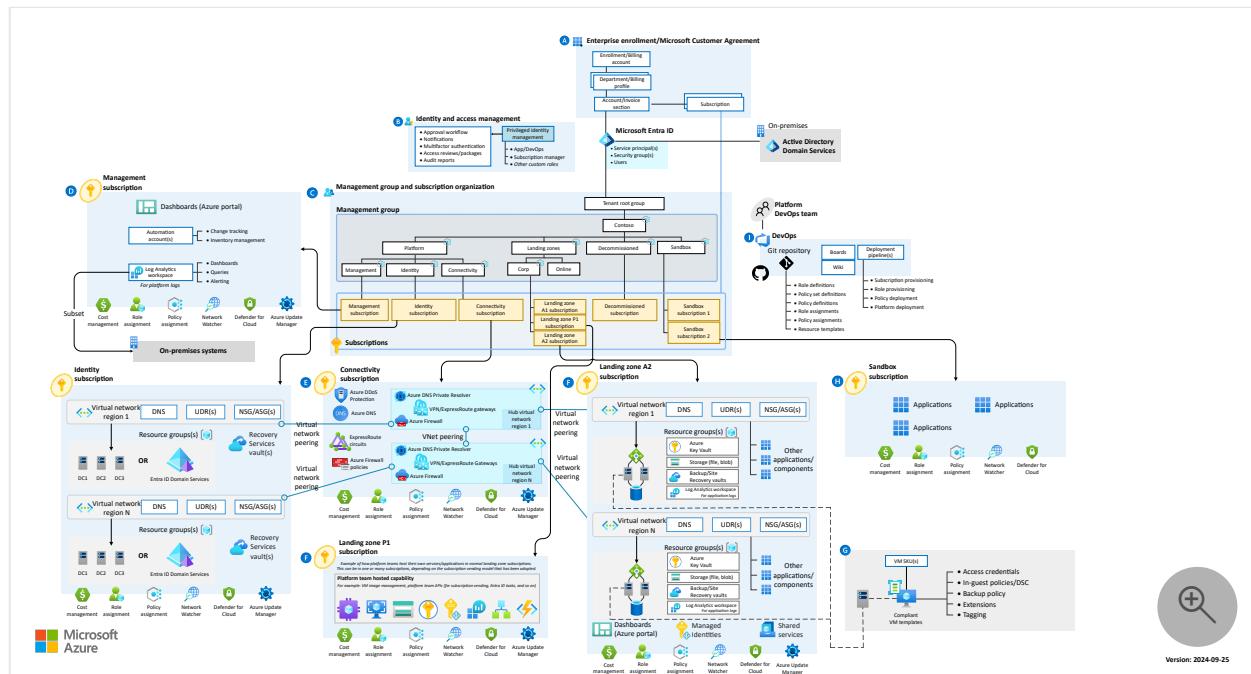
 Yes

 No

Azure landing zone design areas and conceptual architecture

Article • 09/27/2024

The Azure landing zone conceptual architecture below is an example of scaled-out target architecture intended to help organizations operate successful cloud environments that drive their business while maintaining best practices for security and governance. Each Azure landing zone implementation option provides a deployment approach and defined design principles. Learn about these design areas before choosing an implementation option. Use this architecture as a starting point. Download the [Visio file](#) and modify it to fit your specific business and technical requirements when planning your landing zone implementation.



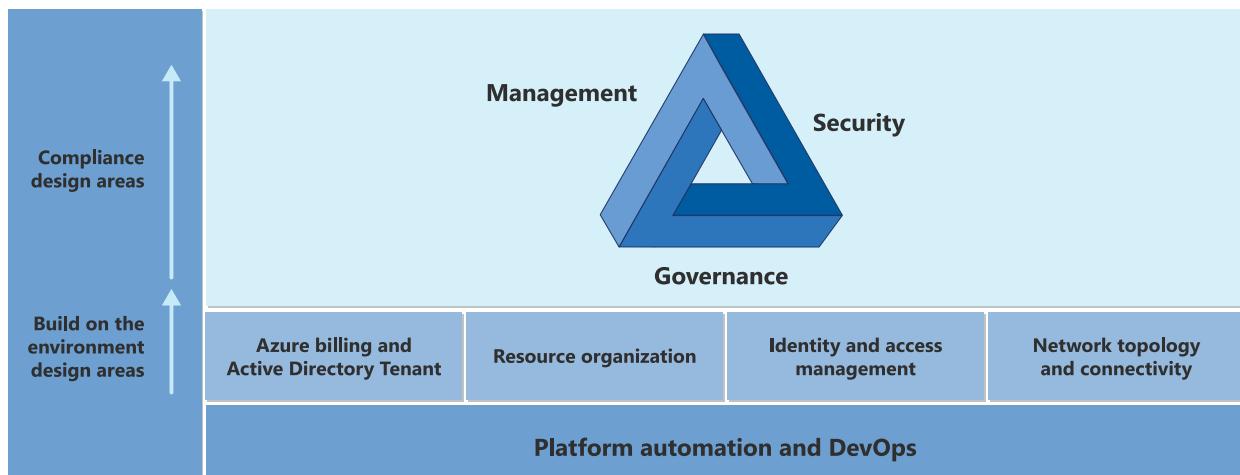
Environment design areas

Whatever the deployment option, you should carefully consider each design area. Your decisions affect the platform foundation on which each landing zone depends. You can follow design area concepts indicated with the letters "A" through "I" in the design area diagrams to illustrate the hierarchy of resource organization in the conceptual architecture.

[Expand table](#)

Design area	Objective
Azure billing and Microsoft Entra tenant	Proper tenant creation, enrollment, and billing setup are important early steps.
Identity and access management	Identity and access management is a primary security boundary in the public cloud. It's the foundation for any secure and fully compliant architecture.
Network topology and connectivity	Networking and connectivity decisions are an equally important foundational aspect of any cloud architecture.
Resource organization	As cloud adoption scales, considerations for subscription design and management group hierarchy have an impact on governance, operations management, and adoption patterns.

Compliance design areas



Security, governance, and compliance are key topics when designing and building an Azure environment. These topics help you start from strong foundations and ensure that solid ongoing processes and controls are in place.

The tools and processes you implement for managing environments play an important role in detecting and responding to issues. These tools work alongside the controls that help maintain and demonstrate compliance.

As the organization's cloud environment develops, these compliance design areas are the focus for iterative refinement. This refinement might be because of new applications that introduce specific new requirements, or the business requirements changing. For example, in response to a new compliance standard.

[] Expand table

Design area	Objective
Security	Implement controls and processes to protect your cloud environments.
Management	For stable, ongoing operations in the cloud, a management baseline is required to provide visibility, operations compliance, and protect and recover capabilities.
Governance	Automate auditing and enforcement of governance policies.
Platform automation and DevOps	Align the best tools and templates to deploy your landing zones and supporting resources.

Design area process

These design areas describe what to consider before deploying a landing zone. Together, they establish a process to aid in exploring otherwise complex topics. These topics are typically involved in making critical decisions about your environment. Evaluate each design area to help you understand any changes you might need to make to the Azure landing zone implementation options.

Evaluating each of the design areas sequentially provides a process that simplifies the design of any complex environments. If you've already addressed one or more of the design areas to your satisfaction, move on to the next area.

In this process, you're provided with a list of roles or functions that are typically required to make design decisions. You also see a series of considerations, recommendations, and scope boundaries to help shape the discussion and decision-making process.

Next steps

You can implement these design areas over time so that you can grow into your cloud operating model. Review the methodologies related to each of the design areas to understand in more detail the considerations and decisions required to implement a landing zone.

Within each design area, you find considerations to help shape your internal discussions and recommendations. These considerations provide specific guidance to help align your journey to the Azure landing zone conceptual architecture.

Alternately, there are rich, opinionated implementation options that start with a defined position on each design area.

With an understanding of the modular design areas, your next step is to choose the landing zone implementation option that best aligns with your cloud adoption plan and requirements.

[Choose an implementation option](#)

Feedback

Was this page helpful?

 Yes

 No

Azure billing offers and Microsoft Entra tenants

Article • 11/28/2024

This critical design area focuses on the two highest levels of alignment across all of your Azure deployments; your Azure billing offer and the association of that offer with a Microsoft Entra tenant.

Design area review

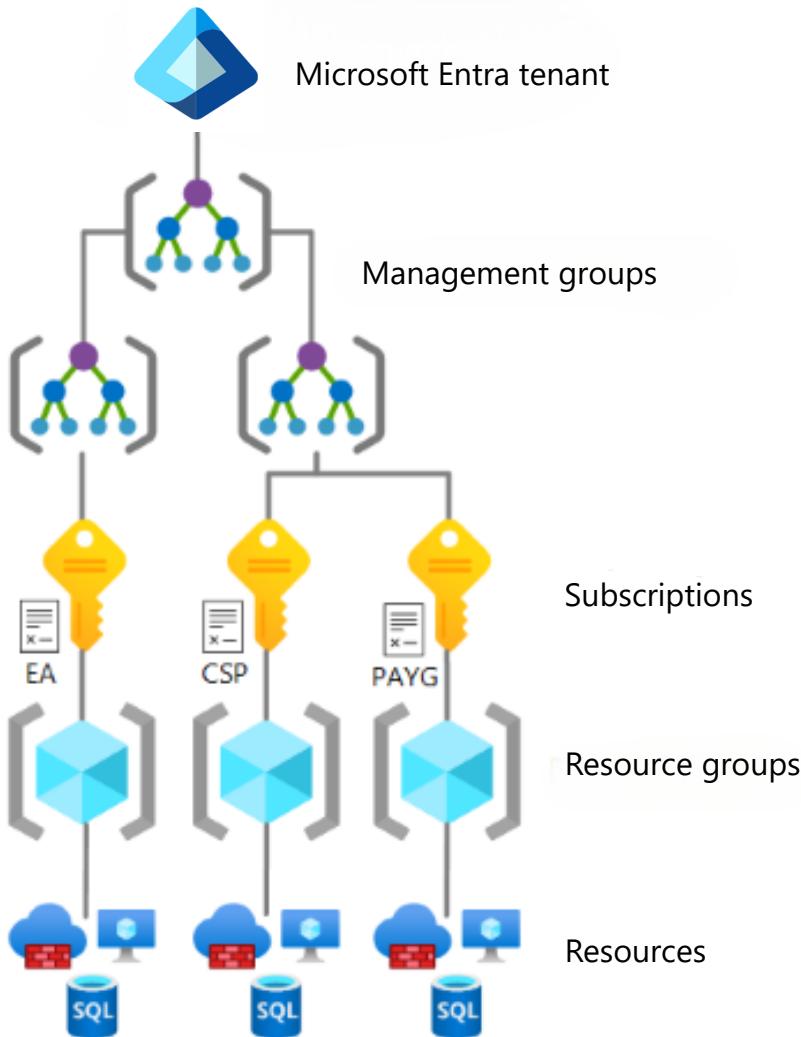
Involved roles or functions: This design area will likely require support from one or more of the following functions or roles to make decisions and implement those decisions: [cloud strategy](#), [cloud platform](#), and [cloud center of excellence](#)

Scope: The objective of this exercise is to evaluate the various [offer types](#) and Microsoft Entra tenant association is best suited for your overall environment.

Out of scope: This design area doesn't focus on the identity or management aspects of Microsoft Entra ID, only the tenant in which your identities will ultimately be hosted. That guidance will be reviewed in the [identity and access management design area](#).

Design area overview

The Azure service presents a range of [active subscription offers](#), and customers can use these offers at the same time to gain flexible billing options. Example subscriptions include Enterprise Agreement (Enterprise Agreement), Microsoft customer agreement, cloud service provider, and others.



The Azure landing zone architecture supports subscriptions from any [Azure offer](#). Subscriptions can only exist within one Microsoft Entra tenant to then relocate into the management group hierarchy within that tenant. They can then be managed by the various controls with enterprise-scale platforms like Azure Policy and role-based access control (RBAC).

Note

The Azure landing zone accelerator implementation is scoped and deployed to one Microsoft Entra tenant; however, billing options can span across multiple Microsoft Entra tenants. For example, an Enterprise Agreement enrollment supports Azure subscriptions across different Microsoft Entra tenants.

Explore the further information, considerations and recommendations for each approach below:

- [Enterprise Agreement \(EA\)](#)
- [Microsoft customer agreement \(MCA\)](#)

- Cloud solution provider agreement (CSP)
 - Define a Microsoft Entra tenant
-

Feedback

Was this page helpful?

 Yes

 No

Plan for Enterprise Agreement enrollment

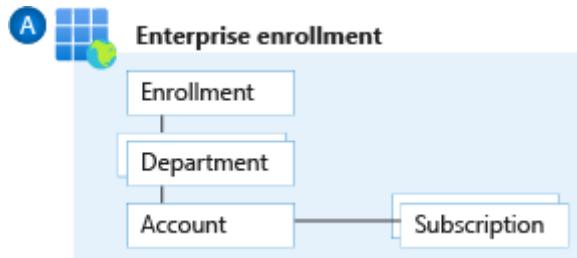
Article • 11/28/2024

Enterprise Agreement enrollment represents the commercial relationship between Microsoft and how your organization uses Azure. It provides billing foundation for your subscriptions and how your digital estate is administered. The Microsoft Cost Management blade in the Azure portal helps you to manage your Enterprise Agreement enrollment. An enrollment often represents an organization's hierarchy, including departments, accounts, and subscriptions. This hierarchy represents cost centers within an organization.

ⓘ Note

The Azure EA portal <https://ea.azure.com> has been retired as of February 15, 2024. Customers should now use the Cost Management blade in the Azure portal to manage their enrollments as documented further in:

- [Azure EA portal administration](#)
- [Get started with your Enterprise Agreement billing account](#)



- Departments help to segment costs into logical groupings and set a budget or quota at the department level. The quota isn't firmly enforced; it's used for reporting purposes.
- Accounts are organizational units in the Cost Management blade of the Azure portal. They can be used to manage subscriptions and access reports.
- Subscriptions are the smallest units in the Azure portal. They're containers for Azure services that are managed by a Service Administrator. This is where your organization deploys Azure services.
- [Enterprise Agreement enrollment roles](#) link users with their functional role. These roles are:

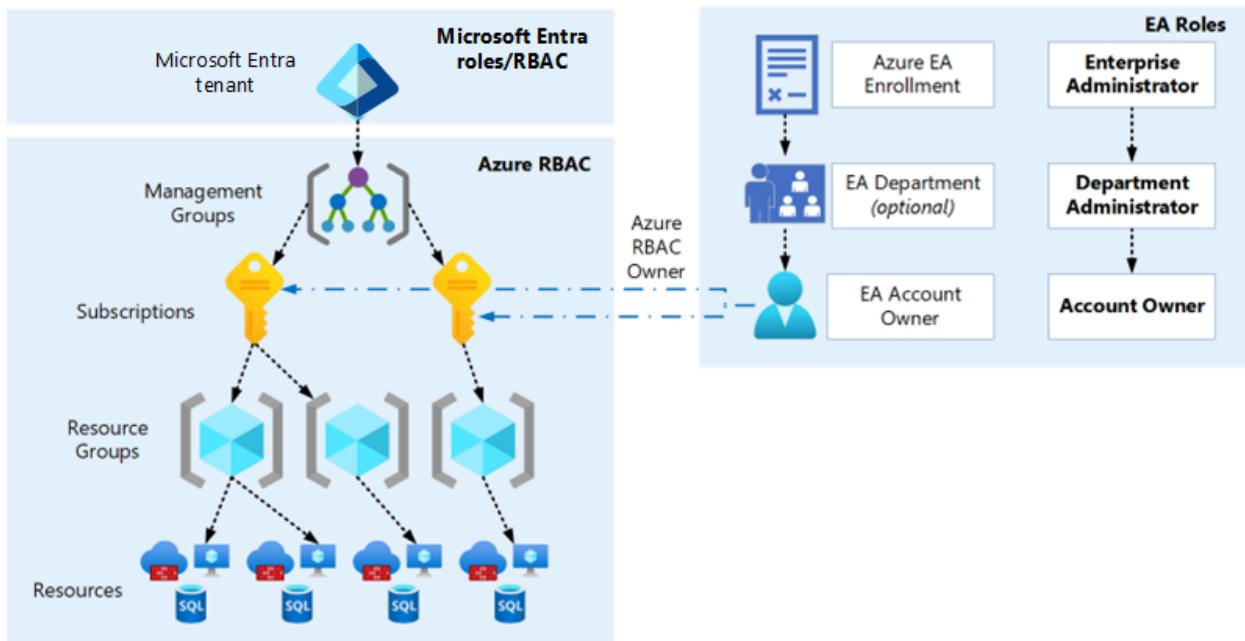
- Enterprise Administrator
- Department Administrator
- Account owner
- Service Administrator
- Notification Contact

How an Enterprise Agreement enrollment relates to Microsoft Entra ID and Azure RBAC

When your organization uses an Enterprise Agreement enrollment for Azure subscriptions, it's important to understand the various authentication and authorization boundaries and the relationship between these boundaries.

There is an inherent trust relationship between Azure subscriptions and a Microsoft Entra tenant, which is described further in [Associate or add an Azure subscription to your Microsoft Entra tenant](#). An Enterprise Agreement enrollment can also use a Microsoft Entra tenant as an identity provider, depending on the [authentication level](#) set on the enrollment and which option was selected when the enrollment account owner was created. However, apart from the account owner, Enterprise Agreement enrollment roles don't provide access to Microsoft Entra ID or the Azure subscriptions within that enrollment.

For example, a finance user is granted an Enterprise Administrator role on the Enterprise Agreement enrollment. They're a standard user without elevated permissions or roles assigned to them in Microsoft Entra ID or on any Azure management group, subscription, resource group, or resource. The finance user can only perform the roles listed at [Managing Azure Enterprise Agreement roles](#) and can't access the Azure subscriptions on the enrollment. The only Enterprise Agreement role with access to Azure subscriptions is the account owner because this permission was granted when the subscription was created.



Design considerations

- The enrollment provides a hierarchical organizational structure to govern how subscriptions are managed. For more information, see [Managing Azure Enterprise Agreement roles](#).
- A range of administrators can be assigned to a single enrollment.
- Each subscription should have a designated account owner. See [Azure EA administration guide](#) for details on how to change this, if needed.
- Each account owner is a subscription owner for any subscriptions provisioned under that account.
- A subscription can belong to only one account at a time.
- A specific set of criteria can be used to determine whether a subscription should be suspended.
- Departments and accounts can filter enrollment billing and usage reports.
- Review [Programmatically create Azure Enterprise Agreement subscriptions with the latest APIs](#) for more information about Enterprise Agreement subscription limitations.

⚠️ Warning

You will not be able to create new subscriptions or transfer existing subscriptions from an enrollment account if the associated UPN is deleted from Microsoft Entra

Design recommendations

- Only use the authentication type `Work or school account` for all account types. Avoid using the `Microsoft account (MSA)` account type.
- Set up a Notification Contact email address to ensure notifications are sent to an appropriate group mailbox.
- An organization can have various structures, including functional, divisional, geographic, matrix, or team structures. Using departments and accounts to map your organization's structure to your enrollment hierarchy can help with separating billing.
- Use [Cost Management](#) reports and views, which can use Azure metadata (for example, tags and location) to explore and analyze your organization's costs.
- Restrict and minimize the number of account owners within the enrollment to limit administrator access to subscriptions and associated Azure resources.
- Assign a budget for each department and account, and establish an alert associated with the budget.
- Create new departments for IT only if the corresponding business domains have independent IT capabilities.
- If you use multiple Microsoft Entra tenants, verify that the account owner is associated with the same tenant as where subscriptions for the account are provisioned.
- For development/testing (dev/test) workloads, use the [Enterprise Dev/Test](#) offer, where available. Ensure you comply with the [terms of use](#).
- Don't ignore notification emails sent to the notification account email address. Microsoft sends important Enterprise Agreement prompts to this account.
- Don't move, rename or delete the Entra ID user associated to EA enrollment account.
- Periodically audit the Cost Management blade in the Azure portal to review who has access, and when possible, avoid using a Microsoft account.

- Enable both DA View Charges and AO View Charges on every Enterprise Agreement enrollment to allow users with the correct permissions to view Cost Management data.
 - Any user that has permissions upon an enrollment to create subscriptions, as detailed [here](#), must be protected with multifactor authentication as any other privileged account should be as documented [here](#)
-

Feedback

Was this page helpful?

 Yes

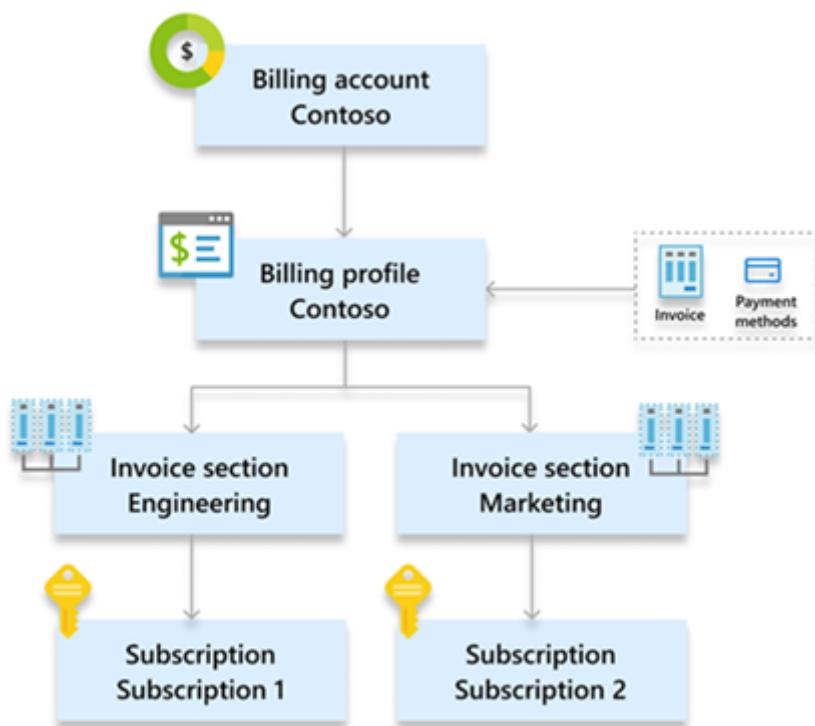
 No

Plan for the Microsoft customer agreement service

Article • 11/28/2024

The [Microsoft customer agreement](#) is a recent and modern Azure service and commerce platform. It represents the commercial relationship between Microsoft and how your organization uses Azure. The agreement enables a streamlined, electronic transaction in an 11-page agreement that doesn't expire. It provides a billing foundation for your subscriptions and affects how your digital estate is administered. You can manage your agreement in the [Azure portal](#).

The Microsoft customer agreement often represents an organization's hierarchy, which consists of billing profiles, invoice sections, and subscriptions. This hierarchy represents cost centers within an organization.



ⓘ Important

If migrating from an Enterprise Agreement to a Microsoft customer agreement, review the following articles:

- [Complete Enterprise Agreement tasks in your billing account for a Microsoft customer agreement](#)

- [Set up your billing account for a Microsoft customer agreement](#)

Design considerations

- The agreement provides a hierarchical organizational structure to govern how subscriptions are managed. For more information, see [Organize costs by customizing your billing account](#).
- An agreement billing account is managed by a single Microsoft Entra tenant. However, subscriptions across different Microsoft Entra tenants are supported by a single agreement. For more information, see [How tenants and subscriptions relate to billing account](#) and [Manage subscriptions under multiple tenants in a single Microsoft customer agreement](#).
- New Azure subscriptions provisioned with an agreement are associated with the Microsoft Entra tenant in which the agreement billing account is located.
- Agreements use the role-based access control (RBAC) model. Multiple users can be assigned with the required roles at the same scopes (for example, billing account, billing profile, and invoice section). These billing roles and assignments are outside of standard Azure RBAC roles and assignments. They can't be assigned at a management group or resource group scope.
- A subscription can belong to only one invoice section at any time. Subscriptions can only be moved between invoice sections within the same billing profile.
- An optional purchase order number can be set up on a billing profile.
- A specific set of criteria can be used to determine whether a subscription should be suspended.
- Before you provision more billing profiles, [review the potential impact to charges and reservations](#).
- Use [Microsoft Cost Management](#) reports and views, which explore and analyze your organization's costs with Azure metadata.

Design recommendations

- Set up a Notification Contact email address on the agreement billing account to ensure notifications are sent to an appropriate group mailbox.

- Assign a budget for each invoice section or billing profile, and establish an alert associated with the budget.
- An organization can have various structures, such as functional, divisional, geographic, matrix, or team. Use organizational structures to map your organization to your agreement hierarchy. Invoice sections are suitable for most scenarios.
- If your business domain has independent IT capabilities, create a new invoice section for IT.
- Don't ignore notifications sent to the contact email address. Microsoft sends important prompts to this address.
- Periodically audit the agreement billing RBAC role assignments to review who has access.
- For development/testing (dev/test) workloads, use the Microsoft Azure plan for dev/test offer, where available. Ensure you comply with the [terms of use](#).
- Any user with [permissions](#) on an invoice section, billing profile, or billing account to create subscriptions must use multifactor authentication (MFA). This applies to all privileged accounts as documented [here](#).

Feedback

Was this page helpful?



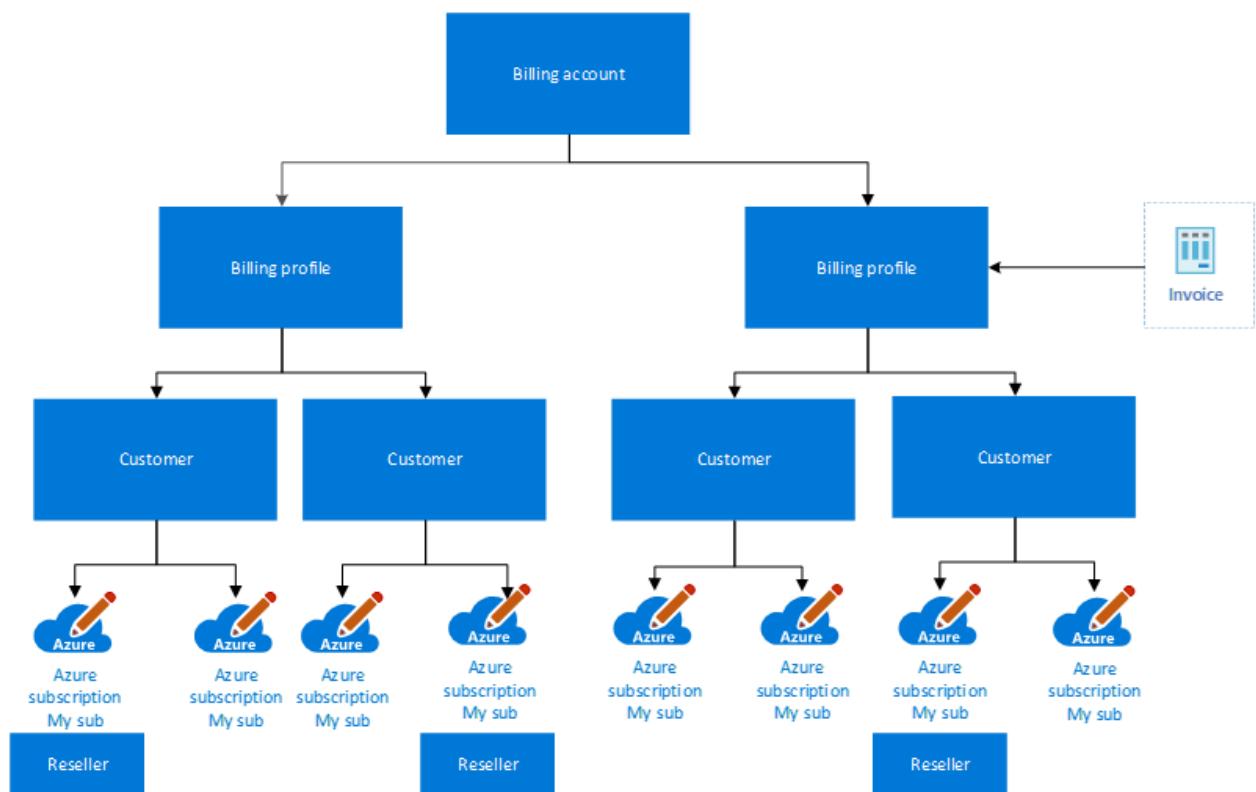
Plan for the Cloud Solution Provider service

Article • 11/29/2024

The Cloud Solution Provider (CSP) service gives Microsoft partners access to Microsoft cloud services within one platform. It supports partners to:

- Own the customer lifecycle and end-to-end relationship.
- Set pricing, terms, and directly bill customers.
- Directly provision and manage subscriptions.
- Attach services that add value.
- Be the customer's first point of contact for support.

[Azure in CSP](#) is an Azure plan with various subscriptions that are hosted by the partner's [Microsoft Partner Agreement \(MPA\)](#). The MPA is similar to the Microsoft customer agreement. Both are hosted on the modern commerce platform and use a [simplified purchase agreement](#).



ⓘ Important

The partner CSP completely manages an MPA.

Design considerations

- A [CSP reseller relationship](#) must exist between the partner and each Microsoft Entra tenant in which the customer wants to provision a new Azure plan and CSP subscriptions.
- Only the partner can provision an Azure plan and CSP subscriptions.
- A specific set of criteria can be used to determine whether a subscription should be suspended; a partner can also suspend a subscription.
- The partner can allow customers to view their Azure usage fees on a per customer basis. For more information, see [Enable the policy to view Azure usage charges](#). Partners can also use other tools to provide customers with access to their charges.
- By default, only the partner Azure Reservations can be purchased by the partner for their customer. However, the [Customer Permissions feature](#) grants customers permission to purchase Azure Reservations from their CSP.

Design recommendations

- Work with your CSP partner to ensure that Azure Lighthouse is used for administration on behalf of (AOBO) access for most support scenarios. For more information, see [Azure Lighthouse and the Cloud Solution Provider program](#).
- Partners, should use, or migrate, to [granular delegated admin privileges \(GDAP\)](#) instead of utilizing delegated admin privileges (DAP).
- Follow and implement the [Customer security best practices](#)
- Partners should follow and implement the [CSP security best practices](#)
- Work with your CSP partner to understand how to create support cases and escalation processes.
- Discuss how to create self-service subscriptions with your CSP partner.
- Use [Microsoft Cost Management](#) reports and views. These reports can use Azure metadata, like tags and location, to explore and analyze your organization's costs.
- Any user that has permissions upon an invoice section, billing profile or billing account to create subscriptions, as detailed [here](#), must be protected with multifactor authentication (MFA) as any other privileged account should be as documented [here](#)

Next steps

Learn how to improve your security posture by defining your Microsoft Entra tenants.

[Define Microsoft Entra tenants](#)

Feedback

Was this page helpful?

 Yes

 No

Define Microsoft Entra tenants

Article • 11/28/2024

A Microsoft Entra tenant provides identity and access management, which is an important part of your security posture. A Microsoft Entra tenant ensures that authenticated and authorized users only access the resources to which they have permissions. Microsoft Entra ID provides these services to applications and services deployed in and outside of Azure (such as on-premises or third-party cloud providers).

Microsoft Entra ID is also used by software as a service (SaaS) applications such as Microsoft 365 and Azure Marketplace. Organizations already using on-premises AD can integrate it with their current infrastructure and extend cloud authentication. Each Microsoft Entra directory has one or more domains. A directory can have many subscriptions associated with it but only one Microsoft Entra tenant.

Ask basic security questions during the design phase, such as how your organization manages credentials and how it controls human, application, and programmatic access.

💡 Tip

If you have multiple Microsoft Entra tenants, review [Azure landing zones and multiple Microsoft Entra tenants](#) and its associated content.

Design considerations:

- An Azure subscription can only trust one Microsoft Entra tenant at a time, further information can be found at [Associate or add an Azure subscription to your Microsoft Entra tenant](#)
- Multiple Microsoft Entra tenants can function in the same enrollment. Review [Azure landing zones and multiple Microsoft Entra tenants](#)
- Azure Lighthouse only supports delegation at the subscription and resource group scopes.
- The `*.onmicrosoft.com` domain name created for each Microsoft Entra tenant must be globally unique as per the [terminology section in what is Microsoft Entra ID?](#)
 - The `*.onmicrosoft.com` domain name for each Microsoft Entra tenant cannot be changed once created.
- Review [Compare self-managed Active Directory Domain Services, Microsoft Entra ID, and managed Microsoft Entra Domain Services](#) to fully understand the

differences between all the options and how they relate

- Explore the [authentication methods offered by Microsoft Entra ID](#) as part of your Microsoft Entra tenant planning
- If using [Azure Government](#) review the guidance around Microsoft Entra tenants in [Planning identity for Azure Government applications](#)
- If using Azure Government, Azure China 21Vianet, Azure Germany ([closed on October 29, 2021 ↗](#)) then review [National/Regional clouds](#) for further guidance around Microsoft Entra ID

Design recommendations:

- Add one or more custom domains to your Microsoft Entra tenant as per [Add your custom domain name using the Microsoft Entra admin center](#)
 - Review [Microsoft Entra UserPrincipalName population](#) if planning to or using Microsoft Entra Connect to ensure custom domain names are reflected in your on-premises Active Directory Domain Services environment.
- Define your Azure single sign-on strategy, using Microsoft Entra Connect, based on one of the supported [topologies](#).
- If your organization doesn't have an identity infrastructure, start by implementing a Microsoft Entra-only identity deployment. Deployment with [Microsoft Entra Domain Services](#) and [Microsoft Enterprise Mobility + Security](#) provides end-to-end protection for SaaS applications, enterprise applications, and devices.
- [Microsoft Entra multifactor authentication](#) provides another layer of security and authentication. For more security, also enforce [conditional access policies](#) for all privileged accounts.
- Plan for [emergency access](#) or break-glass accounts to prevent tenant-wide account lockout.
- Use [Microsoft Entra Privileged Identity Management](#) to manage identities and access.
- Send all Microsoft Entra diagnostic logs to a central Azure Monitor Log Analytics workspace following the guidance here: [Integrate Microsoft Entra logs with Azure Monitor Logs](#)
- Avoid creating multiple Microsoft Entra tenants. For further information, see [Testing approach for enterprise-scale](#).

- Use [Azure Lighthouse](#) to grant third parties/partners access to Azure resources in customer Microsoft Entra tenants and centralized access to Azure resources in multitenant Microsoft Entra architectures.
-

Feedback

Was this page helpful?



Yes



No

Azure landing zones and multiple Microsoft Entra tenants

Article • 11/28/2024

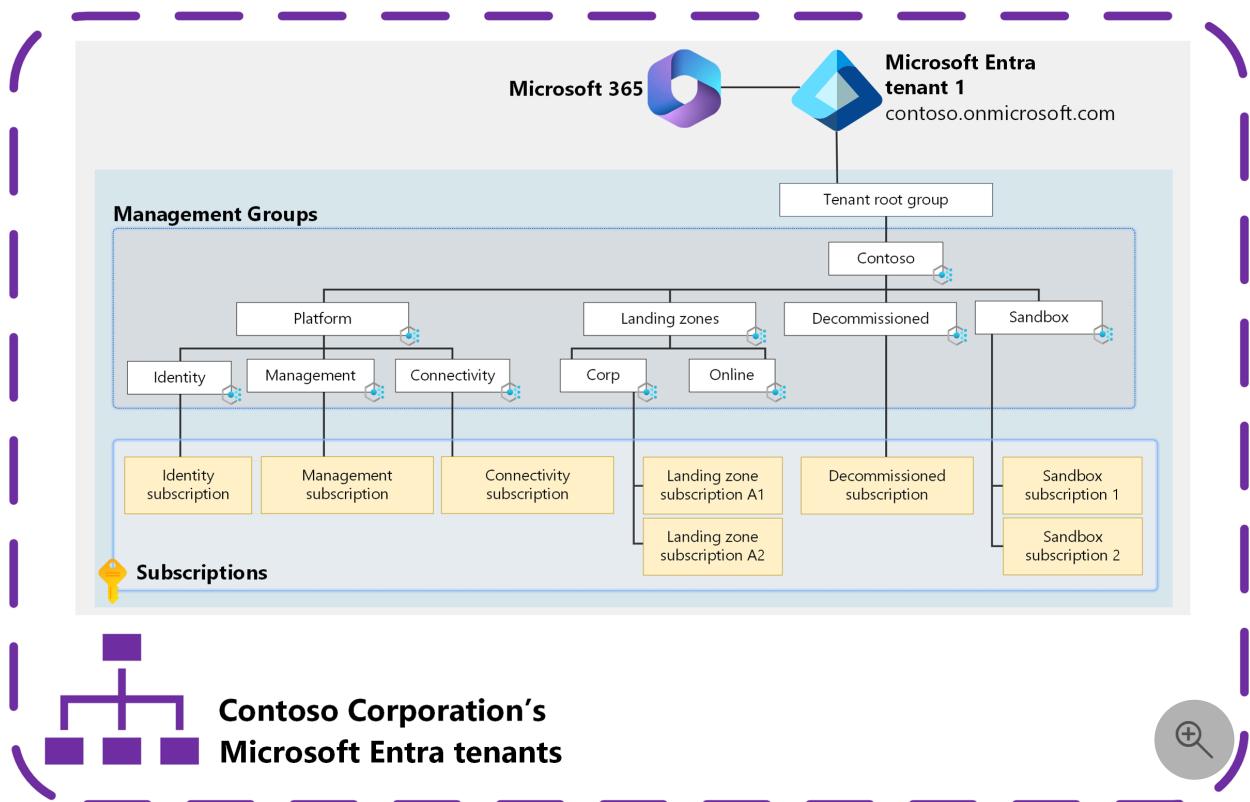
Azure landing zones are built on [management groups](#). [Azure policies](#) are assigned and subscriptions are placed into management groups to provide the required governance controls that an organization needs to meet its security and compliance needs.

💡 Tip

See [Security control mapping with Azure landing zones](#) to learn how to use Azure landing zone and Azure Policy to help achieve your organization's security, compliance, and regulatory needs.

These resources are deployed within a single Microsoft Entra tenant. Management groups and most other Azure resources, like Azure Policy, only support operating within a single Microsoft Entra tenant. An Azure subscription relies on a Microsoft Entra tenant to authenticate users, services, and devices against Azure Resource Manager (ARM) to control plane operations and some Azure services, like Azure Storage, for data plane operations.

Multiple subscriptions can rely on the same Microsoft Entra tenant. Each subscription can only rely on a single Microsoft Entra tenant. For more information, see [Add an existing Azure subscription to your tenant](#).



In the previous diagram, management groups, Azure Policies, and Azure subscriptions are deployed following the [Azure landing zones conceptual architecture](#) within a single Microsoft Entra tenant.

This approach is recommended for most organizations based on their requirements. This approach gives organizations the best collaboration experience possible and allows them to control, govern, and isolate users and resources within a single Microsoft Entra tenant.

Your organization might be required to use multiple Microsoft Entra tenants for many [scenarios](#). See [how to deploy and manage](#) the Azure landing zone deployment into each of these tenants and [considerations and recommendations](#) for handling multiple Microsoft Entra tenants.

ⓘ Note

This article focuses on Azure, not Microsoft 365 or other Microsoft Cloud offerings, such as Dynamics 365 or Power Platform.

It focuses on [the platform rather than applications](#) that are built on top of the platform in tenants. For information about multiple Microsoft Entra tenants and application architecture, see:

- [Multitenant apps in Microsoft Entra ID](#)
- [Architect multitenant solutions on Azure](#)

Why a single Microsoft Entra tenant is sufficient

There are reasons you might require multiple Microsoft Entra tenants, but it's important to understand why a single Microsoft Entra tenant is typically sufficient. It should be the default starting point for all organizations.

Use your existing corporate Microsoft Entra tenant for Azure subscriptions for the best productivity and collaboration experience across the platform.

Within a single tenant, development teams and application owners can have the least privileged roles to create non-production instances of Azure resources and trusted apps, test apps, test users and groups, and test policies for those objects. For more information about how to delegate administration with a single tenant, see [Resource isolation in a single tenant](#).

Only create more Microsoft Entra tenants when there are requirements that can't be met by using the corporate Microsoft Entra tenant.

With Microsoft 365, the corporate Microsoft Entra tenant is generally the first tenant provisioned in the organization. This tenant is used for corporate application access and Microsoft 365 services. It supports the collaboration within an organization. The reason to start with this existing tenant is because it's already been provisioned, managed, and secured. The defined lifecycle of the identities is likely already established. This course makes the task of onboarding new apps, resources, and subscriptions easier. It's a mature, understood environment with established process, procedures, and controls.

Complexities with multiple Microsoft Entra tenants

When you create a new Microsoft Entra tenant, it requires extra work to provision, manage, secure, and govern the identities. You must also establish the required policies and procedures. Collaboration is best in a single Microsoft Entra tenant. Moving to a multitenant model creates a boundary, which can result in user friction, management overhead, and increase the attack surface area, which can cause a security risk and complicates product scenarios and limitations. Some examples include:

- **Multiple identities for users and administrators for each tenant** – If [Microsoft Entra B2B](#) isn't used, the user has multiple sets of credentials to manage. For more information, see [Considerations and recommendations for multitenant Azure landing zone scenarios](#).
- **Azure services limitations in supporting multiple Microsoft Entra tenants** – Azure workloads that only support identities homed in the tenant to which it's bound to.

For more information, see [Azure products and services Microsoft Entra integration](#).

- **No centralized configuration or management for Microsoft Entra tenants –** Multiple security policies, management policies, configuration, portals, APIs, and JML (joiners, movers, and leavers) processes.
- **Billing and licensing complexities and potential requirement for license duplication for Microsoft Entra ID P1 or P2 licenses** - For more information, see [Considerations and recommendations for multitenant Azure landing zone scenarios](#).

Organizations need to be clear about why they're deviating from the corporate Microsoft Entra tenant model to ensure the extra overhead and complexity is justified in meeting the requirements. There are examples of these instances in the [scenarios article](#).

ⓘ Important

Use [Microsoft Entra Privileged Identity Management](#) to protect privileged roles within Microsoft Entra ID and Azure.

The ownership of privileged roles across internal teams and departments can provide a challenge as the Identity team and the Azure team are often in different teams, departments, and organization structures.

The teams that operate Azure are responsible for Azure services and want to ensure the security of the services that they manage. When individuals outside of that team have roles with the power to potentially access their environments, the security is weaker. For more information, see [Understand required cloud functions](#).

Microsoft Entra ID provides controls that help mitigate this problem on a technical level, but this issue is also a people and process discussion. For more information, see [Recommendations](#).

ⓘ Important

Multiple Microsoft Entra tenants are **not the recommended** approach for most customers. A single Microsoft Entra tenant, typically the corporate Microsoft Entra tenant, is recommended for most customers because it provides the necessary separation requirements.

For more information, see:

- [Define Microsoft Entra tenants](#)
- [Testing approach for Azure landing zones](#)

- [Introduction to delegated administration and isolated environments](#)
- [Resource isolation in a single tenant](#)
- [Your Microsoft 365 for enterprise tenants](#)

Next steps

[Scenarios for multiple Microsoft Entra tenants](#)

Feedback

Was this page helpful?

 Yes

 No

Scenarios for multiple Microsoft Entra tenants

Article • 11/28/2024

There are a few reasons why an organization might need, or might want to investigate, multiple Microsoft Entra tenants. The most common scenarios are:

- Mergers and acquisitions
- Regulatory or country/region compliance requirements
- Business unit or organizational isolation and autonomy requirements
- Independent software vendor (ISV) delivering SaaS applications from Azure
- Tenant level testing / Microsoft 365 testing
- Grassroots / Shadow IT / start-ups

Mergers and acquisitions

As organizations grow over time, they might acquire other companies or organizations. These acquisitions are likely to have existing Microsoft Entra tenants already established that host and provide services, such as Microsoft 365 (Exchange Online, SharePoint, OneDrive, or Teams), Dynamics 365, and Microsoft Azure, to the company or organization.

Typically, in an acquisition, the two Microsoft Entra tenants are consolidated into a single Microsoft Entra tenant. This consolidation reduces the management overhead, improves the collaboration experience, and presents a single brand identity to other companies and organizations.

Important

A custom domain name (for example, `contoso.com`) can only be associated with one Microsoft Entra tenant at a time. So, consolidating tenants is preferred because a single custom domain name can be used by all identities when a merger or acquisition scenario occurs.

Because of the complexities of consolidating two Microsoft Entra tenants into one, sometimes the tenants are left alone and remain separate for an extended or indefinite period of time.

This scenario can also occur when the organizations or companies want to remain separate because other organizations might acquire their company in the future. If an organization keeps the Microsoft Entra tenants isolated and they don't consolidate them, there's less work if there's a future merger or acquisition of a single entity.

Regulatory or country/region compliance requirements

Some organizations have strict regulatory or country/region compliance controls and frameworks (for example, UK Official, Sarbanes Oxley (SOX), or NIST). Organizations might create multiple Microsoft Entra tenants to meet and comply with these frameworks.

Some organizations that have offices and users around the globe with stricter data residency regulations might also create multiple Microsoft Entra tenants. But this particular requirement is usually addressed within a single Microsoft Entra tenant using features, such as [Microsoft 365 Multi-Geo](#).

Another scenario is when organizations require [Azure Government \(US Government\)](#) or [Azure China \(operated by 21Vianet\)](#). These national Azure cloud instances require their own Microsoft Entra tenants. The Microsoft Entra tenants are solely for that national Azure cloud instance and are used for the Azure subscriptions identity and access management services within that Azure cloud instance.

Tip

For more information about Azure national/regional cloud's identity scenarios, see:

- [Azure Government Identity](#)
- [Azure China Cross-border connectivity and interoperability](#)
- [Microsoft Entra authentication & national/regional clouds](#)

Like in the previous scenarios, if your organization has a regulatory or country/region compliance framework to comply with, you might not require multiple Microsoft Entra tenants as the default approach. Most organizations can comply with the frameworks within a single Microsoft Entra tenant by using features, such as [Privileged Identity Management](#) and [Administrative units](#).

Business unit or organizational isolation and autonomy requirements

Some organizations might have complex internal structures across multiple business units, or they might require a high level of isolation and autonomy between parts of their organization.

When this scenario occurs, and the tools and guidance in [Resource isolation in a single tenant](#) can't provide the required level of isolation, you might have to deploy, manage, and operate multiple Microsoft Entra tenants.

In scenarios like this, it's more common that there are no centralized functions that are responsible for deploying, managing, and operating these multiple tenants. Instead, they're handed over in full to the separated business unit or part of the organization to run and manage. A centralized architecture, strategy, or CCoE style team might still provide guidance and recommendations on best practices that must be configured in the separate Microsoft Entra tenant.

Warning

Organizations that have operational roles and responsibilities creates challenges between teams that operate the organization's Microsoft Entra tenant. Azure should prioritize creating and agreeing upon a clear RACI between the two teams. This action ensures that both teams can work and deliver their services to the organization and provide value back to the business in a timely manner.

Some organizations have cloud infrastructure and development teams that use Azure. The organizations rely on an identity team that has control over the corporate Microsoft Entra tenant for service principal creation or group creation and management. If there isn't an agreed RACI, there's often a lack of process and understanding between the teams, which leads to friction between the teams and across the organization. Some organizations believe that multiple Microsoft Entra tenants is the only way to overcome this challenge.

But multiple Microsoft Entra tenants creates challenges for end users, increases complexity in securing, managing, and governing multiple tenants, and potentially increases licensing costs. Licenses, such as Microsoft Entra ID P1 or P2, don't span multiple Microsoft Entra tenants. Sometimes, [Microsoft Entra B2B](#) usage can alleviate licensing duplication for some features and services. If you plan to use Microsoft Entra B2B in your deployment, review each feature and service's licensing terms and supportability for Microsoft Entra B2B eligibility.

Organizations in this situation should resolve the operational challenges to ensure that teams can work together in a single Microsoft Entra tenant rather than create multiple Microsoft Entra tenants as a workaround.

Independent software vendor (ISV) delivering SaaS applications from Azure

ISVs that deliver their SaaS (software as a service) products to their customers might benefit from having multiple Microsoft Entra tenants for their Azure usage.

If you're an ISV, you might have separation between your corporate Microsoft Entra tenant, including Azure usage, for your business-as-usual activities, such as e-mail, file sharing, and internal applications. You might also have a separate Microsoft Entra tenant where Azure subscriptions host and deliver the SaaS applications that you provide to your end customers. This approach is common and sensible because it protects you and your customers from security incidents.

For more information, see [Independent software vendor \(ISV\) considerations for Azure landing zones](#).

Tenant level testing / Microsoft 365 testing

Some activities and features in the Microsoft Cloud products, services, and offerings can only be tested in a separate Microsoft Entra tenant. Some examples are:

- Microsoft 365 – Exchange Online, SharePoint, and Teams
- Microsoft Entra ID – Microsoft Entra Connect, Microsoft Entra ID Protection Risk Levels, and SaaS applications
- Testing scripts that use the Microsoft Graph API and can effect and make changes to production

When you want to perform testing like the previous scenarios, a separate Microsoft Entra tenant is your only option.

But the separate Microsoft Entra tenant is **not** for hosting Azure subscriptions that contain workloads, regardless of the environment, for example dev/test. Even dev/test environments should instead be contained in your regular "production" Microsoft Entra tenant.



Tip

For information about how to handle testing Azure landing zones and Azure workloads or resources within Azure landing zones environments, see:

- [How do we handle "dev/test/production" workload landing zones in Azure landing zone architecture?](#)
- [Testing approach for Azure landing zones](#)

Grassroots / Shadow IT / Start-ups

If a team wants to innovate quickly, they might create a separate Microsoft Entra tenant to help them move as quickly as possible. They might, intentionally or unintentionally, avoid the central/platform team's process and guidance for getting access to an Azure environment to do their innovation.

This scenario is common in start-ups where they set up their own Microsoft Entra tenant to run, host, and operate the business and services from. It's typically to be expected, but when start-ups are acquired, the extra Microsoft Entra tenant creates a decision point where the acquiring organization's IT teams decide what to do going forward.

For more information about how to navigate this scenario, see the [Mergers and acquisitions](#) and [Independent software vendor \(ISV\) delivering SaaS applications from Azure](#) sections in this article.

Important

We highly recommend platform teams have an easily accessible and efficient process to give teams access to an Azure sandbox subscription or subscriptions that are homed in the corporate or primary Microsoft Entra tenant for the organization. This process prevents Shadow IT scenarios from occurring and prevents challenges in the future for all parties involved.

For more information about sandboxes, see [Management groups guidance within the resource organization design area](#).

Summary

As detailed in the scenarios, there are several reasons why your organization might require multiple Microsoft Entra tenants. But when you create multiple tenants to meet the requirements within these scenarios, it adds complexity and operational tasks to maintain the multiple tenants and potentially adds costs for licensing requirements. For

more information, see [Considerations and recommendations for Azure landing zones in multitenant scenarios](#).

Next steps

[Considerations and recommendations for multitenant Azure landing zone scenarios](#)

Feedback

Was this page helpful?

 Yes

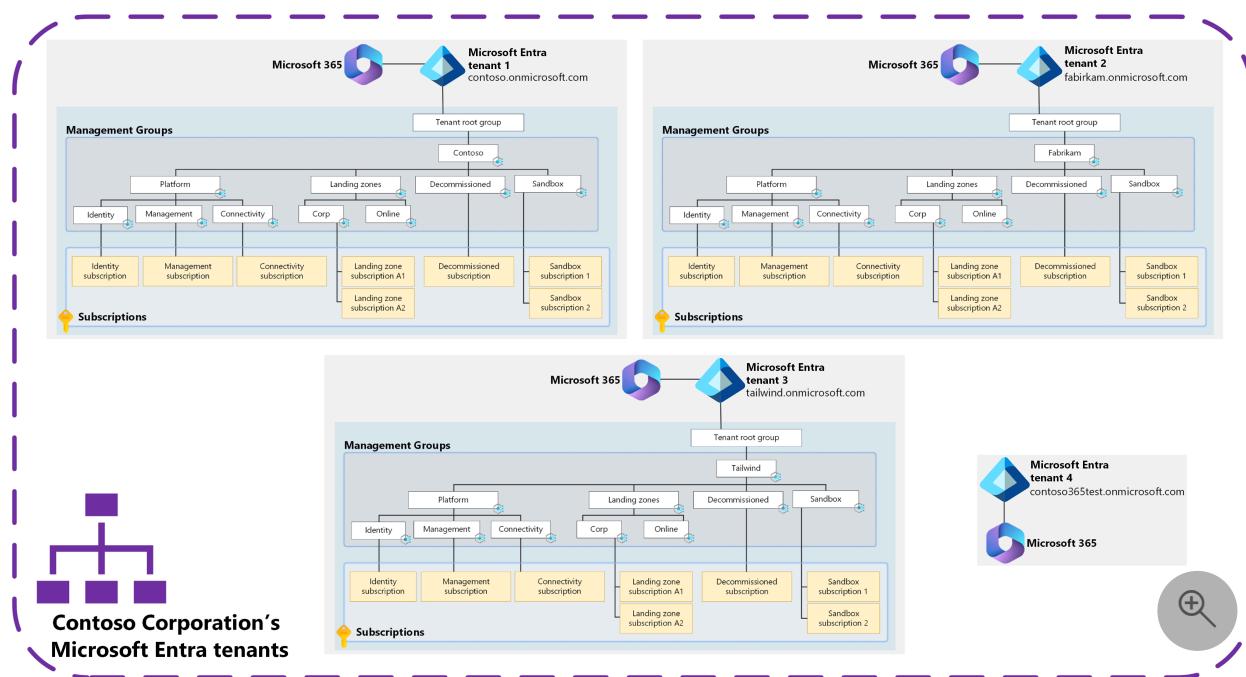
 No

Considerations and recommendations for multitenant Azure landing zone scenarios

Article • 11/29/2024

The article, [Azure landing zones and multiple Microsoft Entra tenants](#), describes how management groups and Azure Policy and subscriptions interact and operate with Microsoft Entra tenants. The article describes the limitation of these resources when they operate within a single Microsoft Entra tenant. Under these conditions, if multiple Microsoft Entra tenants exist, or are required for an organization, the Azure landing zones must be deployed into each of the Microsoft Entra tenants separately.

Azure landing zones with multiple Microsoft Entra tenants



The previous diagram shows an example of the Contoso Corporation, which has four Microsoft Entra tenants due to mergers and acquisitions as the corporation has grown over time.

[Expand table](#)

Microsoft Entra tenant *.onmicrosoft.com domain	Usage notes
contoso.onmicrosoft.com	Primary corporate Microsoft Entra tenant that's used by the Contoso Corporation. Azure and Microsoft 365 services are used in this tenant.
fabrikam.onmicrosoft.com	Primary Microsoft Entra tenant that's used by Fabrikam. Azure and Microsoft 365 services are used in this tenant. This tenant has remained separated since the acquisition by the Contoso Corporation.
tailwind.onmicrosoft.com	Primary Microsoft Entra tenant that's used by Tailwind. Azure and Microsoft 365 services are used in this tenant. This tenant has remained separated since the acquisition by the Contoso Corporation.
contoso365test.onmicrosoft.com	Microsoft Entra tenant that's used by the Contoso Corporation for testing Microsoft Entra ID and Microsoft 365 services and configuration only . All Azure environments live within the contoso.onmicrosoft.com Microsoft Entra tenant.

The Contoso Corporation started out with one Microsoft Entra tenant of `contoso.onmicrosoft.com`. Over time, they made multiple acquisitions of other companies and brought these companies into the Contoso Corporation.

The acquisitions of Fabrikam (`fabrikam.onmicrosoft.com`) and Tailwind (`tailwind.onmicrosoft.com`) brought with them existing Microsoft Entra tenants in which Microsoft 365 (Exchange Online, SharePoint, OneDrive) and Azure services are used within. These companies, and associated Microsoft Entra tenants, are kept separated because parts of the Contoso Corporation and its companies might be sold in the future.

The Contoso Corporation has a separate Microsoft Entra tenant for the sole purpose of testing Microsoft Entra ID and Microsoft 365 services and features. But no Azure services are tested in this separate Microsoft Entra tenant. They're tested in the `contoso.onmicrosoft.com` Microsoft Entra tenant.

💡 Tip

For more information about testing Azure landing zones and Azure workloads and resources within Azure landing zones environments, see:

- [How to handle "dev/test/production" workload landing zones in Azure landing zone architecture](#)
- [Testing approach for Azure landing zones](#)

ⓘ Note

Azure landing zones are deployed within a single Microsoft Entra tenant. If you have multiple Microsoft Entra tenants that you want to deploy Azure resources within, and you want to control, govern, and monitor them by using Azure landing zones, you must deploy Azure landing zones within each of those tenants individually.

Considerations and recommendations for Azure landing zones in multitenant scenarios

This section explains key considerations and recommendations about Azure landing zones and Microsoft Entra multitenant scenarios and usage.

Considerations

- Start with a [single tenant approach](#) to your Microsoft Entra tenant design.
 - The single tenant is typically the organization's corporate Microsoft Entra tenant where the user's identities exist and a service, like Microsoft 365, is running.
 - Only create more Microsoft Entra tenants when there are requirements that can't be met by using the corporate Microsoft Entra tenant.
- Consider using Microsoft Entra ID [administrative units](#) to manage the segregation and isolation of users, groups, and devices (for example, different teams) within a single Microsoft Entra tenant. Use this resource instead of creating multiple Microsoft Entra tenants.
- Consider using sandbox subscriptions for the initial application workload development and investigation. For more information, see [How to handle "dev/test/production" workload landing zones in Azure landing zone architecture](#).
- Migrating Azure subscriptions between Microsoft Entra tenants is complex and requires pre and post migration activities to be completed to enable a migration. For more information, see [Transfer an Azure subscription to a different Microsoft Entra directory](#). It's easier to rebuild the application workload in a new Azure

subscription in the destination tenant. It gives you more control over the migration.

- Consider the [complexities](#) of managing, governing, configuring, monitoring, and securing multiple Microsoft Entra tenants. A single Microsoft Entra tenant is easier to manage, govern, and secure.
- Consider your JML (joiners, movers, and leavers) process, workflows, and tooling. Ensure that these resources can support and handle multiple Microsoft Entra tenants.
- Consider the effect on end users when they manage, govern, and secure multiple identities for themselves.
- When choosing multiple Microsoft Entra tenants, consider the effect on cross-tenant collaboration, especially from an end user's perspective. The Microsoft 365 collaboration experience and support between users within a single Microsoft Entra tenant is optimal.
- Consider the effect on auditing and regulatory compliance checks across multiple Microsoft Entra tenants before choosing an approach.
- Consider the increase in licensing costs when multiple Microsoft Entra tenants are used. Licenses for products like Microsoft Entra ID P1 or P2 or Microsoft 365 services don't span across Microsoft Entra tenants.
- A single Enterprise Agreement enrollment can support and provide subscriptions to multiple Microsoft Entra tenants by setting the authentication level on the enrollment to work and school account cross-tenant. For more information, see [Azure EA portal administration](#).
- A single Microsoft Customer Agreement can support and provide subscriptions to multiple Microsoft Entra tenants. For more information, see [Manage tenants in your Microsoft Customer Agreement billing account](#).
- When opting for a Microsoft Entra multitenant architecture, consider the limitations that might occur for application teams and developers. Be aware of limitations in Microsoft Entra integration for Azure products and services, such as Azure Virtual Desktop, Azure Files, and Azure SQL. For more information, see the [Azure products and services Microsoft Entra integration](#) section in this article.
- Consider using [Microsoft Entra B2B](#) to simplify and enhance user experience and administration when your organization has multiple Microsoft Entra tenants.
- Consider using the Microsoft identity platform, with Microsoft Entra ID with B2B and B2C capabilities, so developers can create applications in a single Azure subscription and within a single tenant. This method supports users from many identity sources. For more information, see [multitenant apps](#) and [Architect multitenant solutions on Azure](#).
- Consider using the features available for multitenant organizations. For more information, see [What is a multitenant organization in Microsoft Entra ID](#).
- Consider [keeping your Azure landing zone up to date](#).

Azure products and services Microsoft Entra integration

Many Azure products and services don't support Microsoft Entra B2B as part of their native Microsoft Entra integration. There are only a few services that support Microsoft Entra B2B authentication as part of their Microsoft Entra integrations. It's safer for the service default to not support Microsoft Entra B2B as part of their Microsoft Entra integration.

Services that provide a native integration with Microsoft Entra ID, such as Azure Storage, Azure SQL, Azure Files, and Azure Virtual Desktop, use a "one-click" or "no-click" style approach to integrate. They require [authentication and authorization](#) scenarios as part of their service. This approach is typically supported against the "home tenant", and some services might enable support for Microsoft Entra B2B/B2C scenarios. For more information about the Azure subscription's relationship to Microsoft Entra ID, see [Associate or add an Azure subscription to your Microsoft Entra tenant](#).

It's important to carefully consider which Microsoft Entra tenant your Azure subscriptions are associated with. This relationship dictates which products and services, and their features, the application or workload teams use that need to support the identities and from which tenant the identities are from. Typically, identities are in the corporate Microsoft Entra tenant.

If multiple Microsoft Entra tenants are used to host all Azure subscriptions, application workload teams can't take advantage of some Azure products and services Microsoft Entra integrations. If the application workload teams have to develop their applications around these imposed limitations, the authentication and authorization process becomes more complex and less secure.

Avoid this problem by using a single Microsoft Entra tenant as the home for all your Azure subscriptions. A single tenant is the best approach for authentication and authorization for your application or service. This simple architecture gives the application workload team less to manage, govern, and control, and it removes potential constraints.

For more information, see [Resource isolation in a single tenant](#).

Recommendations

- Use a single Microsoft Entra tenant, which is usually the corporate Microsoft Entra tenant. Only create more Microsoft Entra tenants when there are requirements that can't be met by using the corporate Microsoft Entra tenant.
- Use sandbox subscriptions to provide application teams safe, controlled, and isolated development environments within the same single Microsoft Entra tenant.

For more information, see [How to handle "dev/test/production" workload landing zones in Azure landing zone architecture](#).

- Use Microsoft Entra multitenant applications when you create integrations from operational tooling, such as ServiceNow, and connect them to multiple Microsoft Entra tenants. For more information, see [Best practices for all isolation architectures](#).
- If you're an ISV, see [Independent software vendor \(ISV\) considerations for Azure landing zones](#).
- Use Azure Lighthouse to simplify cross-tenant management experiences. For more information, see [Azure Lighthouse usage in Azure landing zones multitenant scenarios](#).
- On your Enterprise Agreement enrollments or Microsoft Customer Agreements that are homed in the destination Microsoft Entra tenant, create account owners, invoice section owners, and subscription creators. Assign the owners and creators to the subscriptions they create to avoid having to [change directories on Azure subscriptions](#) once created. For more information, see [Add an account from another Microsoft Entra tenant](#) and [Manage tenants in your Microsoft Customer Agreement billing account](#).
- See the [Microsoft Entra security operations guide](#).
- Keep the number of Global Administrator accounts to a minimum, less than 5 is preferred.
- Enable [Privileged Identity Management \(PIM\)](#) for all admin accounts to ensure no standing privilege and to provide JIT access.
- Require approval in PIM to activate critical roles, such as the Global Administrator role. Consider creating approvers from multiple teams to approve Global Administrator usage.
- Enable monitoring and notifications to all required stakeholders about the Global Administrator role activation.
- Ensure that the "Access management for Azure resources" setting on Global Administrators is set to **No** where it's not required.

Important

Microsoft recommends that you use roles with the fewest permissions. This helps improve security for your organization. Global Administrator is a highly privileged role that should be limited to emergency scenarios when you can't use an existing role.

- Enable and configure the following Microsoft Entra services and features to simplify the multitenant experience for administration and users within your

organization:

- [B2B collaboration](#)
- [B2B direct connect](#)
- [Cross-tenant access settings](#)
- [Cross-tenant synchronization](#)
- [Multitenant Organization](#)
- For organizations with a Microsoft Entra tenant in multiple Microsoft clouds, like Microsoft Azure Commercial cloud, Microsoft Azure China 21Vianet, Microsoft Azure Government, configure [Microsoft cloud settings for B2B collaboration](#) to simplify user's experiences when collaborating across tenants.
- Application teams and developers should review the following resources when constructing applications and services for multi-tenancy:
 - [multitenant apps in Microsoft Entra ID](#)
 - [Architect multitenant solutions on Azure](#)

Next steps

[Automate Azure landing zones across multiple tenants](#)

Feedback

Was this page helpful?

 Yes

 No

Automate Azure landing zones across multiple tenants

Article • 11/28/2024

If your organization has multiple Microsoft Entra tenants with Azure landing zones (ALZ) in each of them, one or multiple times, automation is key. Automation helps to successfully operate and maintain the ALZ deployment at scale across all tenants. There are many approaches to automate ALZ deployments across multiple tenants. The approach you take depends on the reasons your organization has multiple Microsoft Entra tenants.

For example, you might have multiple Microsoft Entra tenants if you're an independent software vendor. It's likely that you want to keep your corporate and SaaS solutions Microsoft Entra tenants separate. The risk of an operation or deployment affecting the other tenant, whether intended or by mistake, reduces.

The following sections provide diagrams and guidance about the approaches that you can take. Choose which approach is best for you based on your requirements, considerations, and recommendations for automating your Azure landing zones deployments when handling multiple Microsoft Entra tenants.

ⓘ Note

Review the following articles first to get an overview of Microsoft Entra tenants:

- [Overview for multiple Microsoft Entra tenants](#)
- [Scenarios for multiple Microsoft Entra tenants](#)
- [Considerations & recommendations for multitenant Azure landing zone scenarios](#)

Approaches

There are two approaches to automate the deployment of Azure landing zones across multiple Microsoft Entra tenants.

[Approach 1 – Complete isolation](#) is the most common approach in multitenant scenarios. This approach keeps the required separation and isolation between Microsoft Entra tenants, which is the most common requirement when using a multitenant approach.

Approach 2 – Shared application registration (multitenant) with multiple service principals is commonly used in Managed Service Provider (MSP) scenarios. In this approach, a deployment stamps pattern can be used to automate the deployment of an almost identical architecture across multiple tenants at scale.

Both of these approaches are provided as examples and inspiration. You can mix and match the approaches in your deployments based on your organization's requirements.

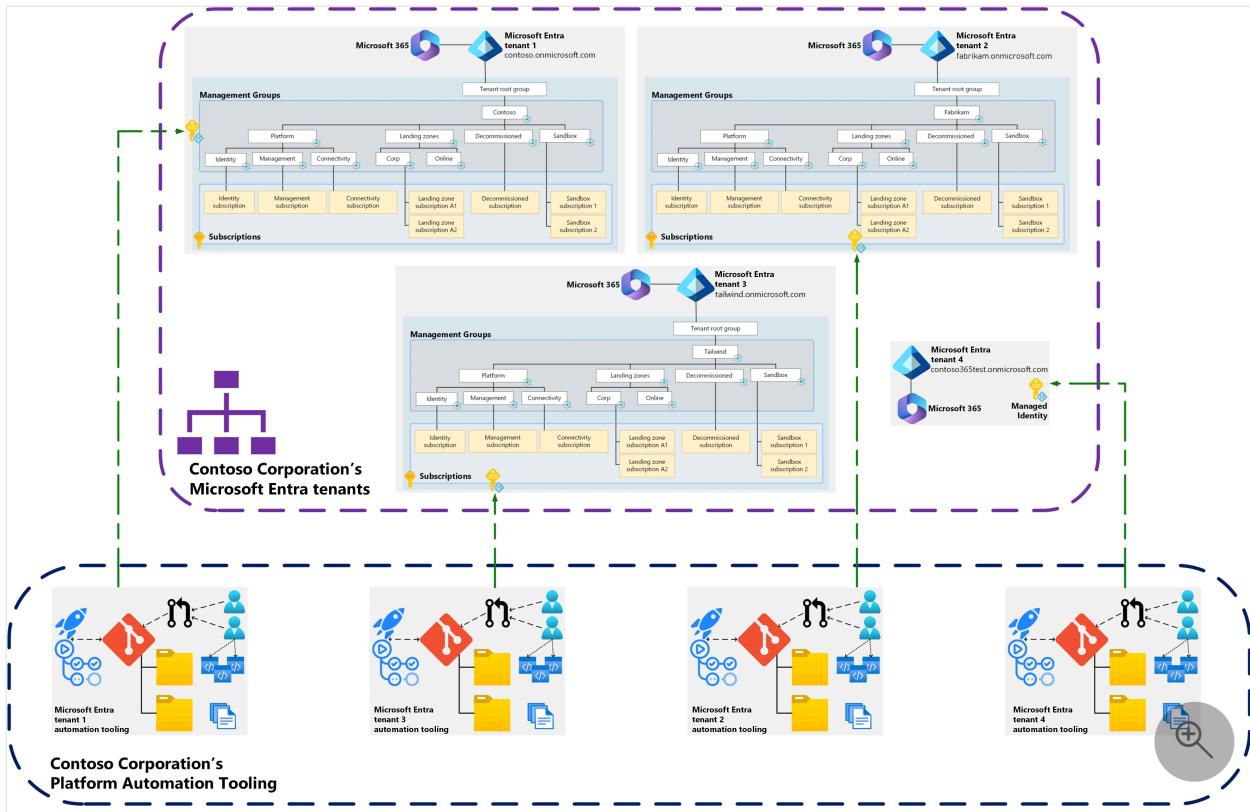
 **Important**

This article covers automating the deployment and operation of Azure landing zones as the platform in each Microsoft Entra tenant that your organization has. The approaches, recommendations, and considerations in this article are **not** intended to be used by application teams that deploy and operate their services and applications into their landing zones (subscriptions). For more information on the different types of landing zones, see [Platform vs. application landing zones](#).

Approach 1 – Complete isolation

In this approach, the primary objective is to keep each Microsoft Entra tenant isolated from each other across all automation components, like:

- A Git repository.
- GitHub Actions or Azure Pipelines (including self-hosted runners, if being utilized).
- Identities that are used for performing tasks from automation, like managed identities assigned to self-hosted runners, service principal names (SPNs), users, or administrators.



In this approach, there are more components to manage that are duplicated per a Microsoft Entra tenant. Some organizations might have regulatory compliance controls enforced on them that mandates this type of segregation and isolation.

! Note

If your organization only allows the use of managed identities for platform automation, you must use this approach or an approach that logs into each tenant individually. Managed identities don't support cross-tenant scenarios. For more information, see [this FAQ](#).

Identities for platform administrators and developers - Approach 1

In this approach, identities should also be isolated in each Microsoft Entra tenant, which means each platform administrator or developer requires a separate user account within each tenant to perform operations within that tenant. These accounts are also used to access the developer tooling, like GitHub or Azure DevOps, for each of the tenants. Carefully consider the effects of administrator and developer productivity following this approach.

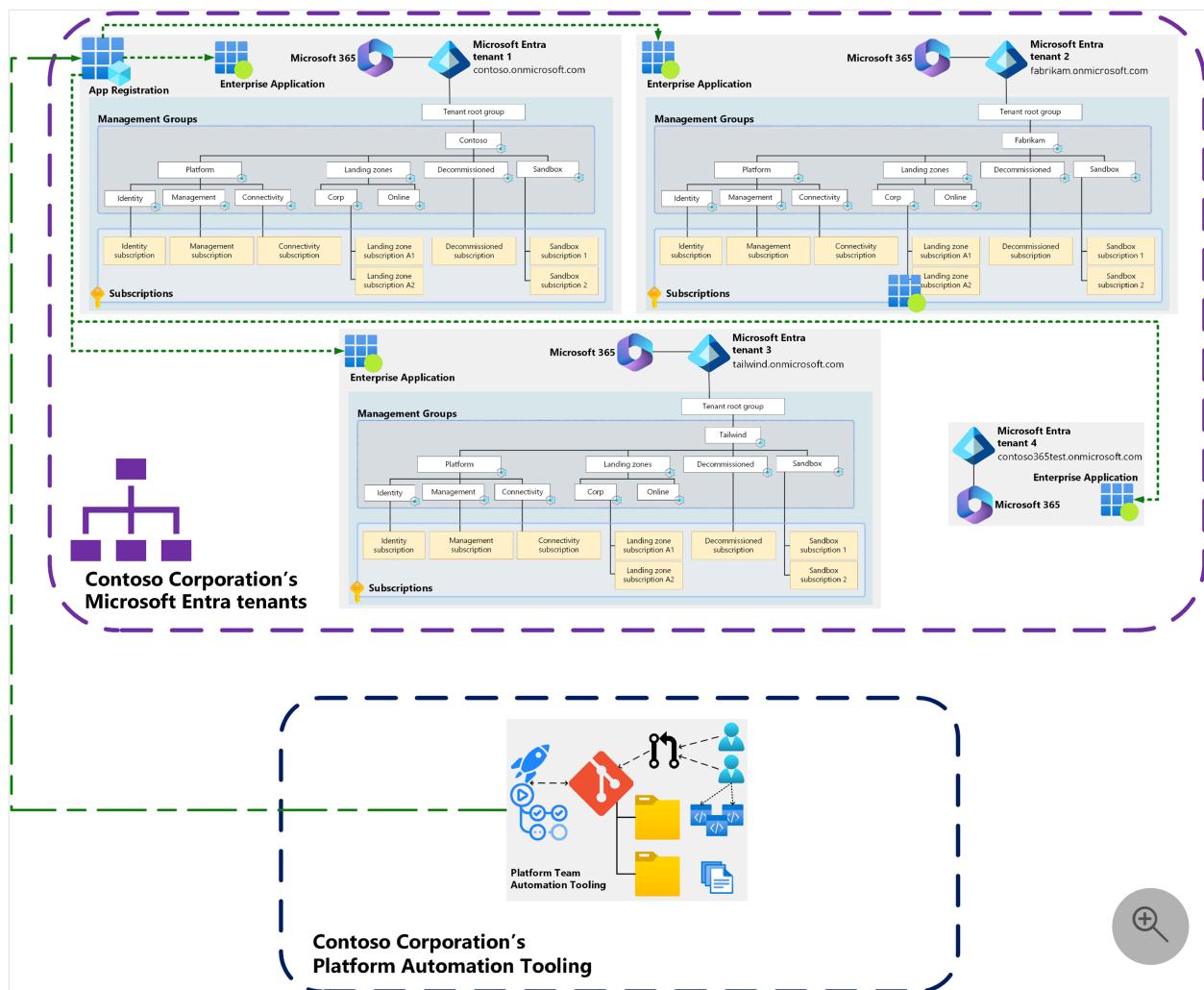
Microsoft Entra B2B and/or Azure Lighthouse can be used, but this option questions the reasoning for having separate Microsoft Entra tenants.

Approach 2 – Shared application registration (multitenant) with multiple service principals

In this approach, an application registration is created in the managing Microsoft Entra tenant. In every Microsoft Entra tenant that you want to manage, a service principal name (SPN) is created in that tenant, based on the application registration. This action allows the workers running the pipeline tasks and steps to sign in to any of the Microsoft Entra tenants with a single set of credentials.

Tip

For information about the relationship between application registrations and enterprise applications (service principles), see [Application and service principal objects in Microsoft Entra ID](#).



Important

In this approach, the single application registration and the associated enterprise applications (service principals) should be monitored for any abnormal activity in

your security information and event management (SIEM) tooling because this is a highly privileged account. It should send alerts and potentially automatically take action, depending on the alert severity.

In the previous example, a single app registration is in the [contoso.onmicrosoft.com](#) Microsoft Entra tenant, and an enterprise application is in each of the Microsoft Entra tenants that's linked to the app registration. This setup allows a pipeline to authenticate and authorize to all the Microsoft Entra tenants by using the single app registration. For more information, see [Making your application multitenant](#).

When you use a centralized pipeline, you might need to build a small mapping table that contains data correlating the Microsoft Entra tenants and other metadata, such as the environment, associated subscriptions, organization name, and identity object ID used for authentication and authorization. This data can be called on during the run of the pipeline in a step that uses some logic and conditions to control which Microsoft Entra tenant it's deployed to and with which identities. The data can be stored in services, such as Azure Cosmos DB or Azure Table storage.

When you handle multiple environments, such as development, test, or production, they can be controlled in the same way by using the same, or separate, application registrations and enterprise applications with pipelines.

You might decide to have separate pipelines for each Microsoft Entra tenant or use a single pipeline. The choice is yours based on your requirements.

Note

Azure Lighthouse works similar to this approach, but Azure Lighthouse doesn't allow the assignment of the RBAC owner, user access administrator, and roles with DataActions permissions. For more information, see [Role support for Azure Lighthouse](#).

The owner and user access roles are typically required in all Azure landing zone deployment scenarios. This requirement removes Azure Lighthouse as an option for the entire platform automation deployment aspect of Azure landing zones, but it's still useful in some scenarios. For more information, see [Azure Lighthouse usage in ALZ multitenant](#).

Identities for platform administrators and developers - Approach 2

In this approach, platform administrators and developers usually only need access to the managing Microsoft Entra tenant. This access grants them access to the developer tooling, like GitHub or Azure DevOps, that all tenants are deployed to and operated from.

They might have access into the other Microsoft Entra tenants via Microsoft Entra B2B or Azure Lighthouse. They use their same account from the managing tenant, or they might have separate accounts, like [the example in the first approach](#).

Next steps

Azure landing zones canary approach with multiple tenants

Feedback

Was this page helpful?

 Yes

 No

Azure landing zones canary approach with multiple tenants

Article • 11/29/2024

When you have multiple Microsoft Entra tenants, you can handle Azure landing zones canary environments the same way you handle them within a single Microsoft Entra tenant. If you're a multitenant ALZ consumer, follow the [canary guidance](#) in each Microsoft Entra tenant separately.

Canary Azure landing zones environment can be independently used to author and test ALZ deployments before you deploy them into the production environment. The term canary is used to avoid confusion with application development environments or test environments. This name is used for illustration purposes only.

Deployment stamps approach

If you're a customer that's following a [deployment stamps pattern](#) for each of your Microsoft Entra tenants ALZ deployments (for example, each of the ALZ deployments is the same in structure, apart from a few naming changes), you might only need to have a single canary deployment because all your ALZ deployments are the same.

Note

This approach is common in a Managed Service Provider (MSP) environment where the MSP manages several Microsoft Entra tenants for different clients that are all similar, except for their naming.

Tip

Customers that follow the deployment stamp style pattern might also benefit from following the [Automation approach 2 – Shared application registration \(multitenant\) with multiple service principals](#).

Next steps

[Azure Lighthouse usage in Azure landing zones multitenant scenarios](#)

Feedback

Was this page helpful?

 Yes

 No

Azure Lighthouse usage in Azure landing zones multitenant scenarios

Article • 11/28/2024

Azure Lighthouse enables multitenant management with scalability, higher automation, and enhanced governance across resources. Azure Lighthouse can be adopted in Azure landing zone scenarios in single or multitenant architectures.

The following considerations and recommendations describe common scenarios for Azure Lighthouse in Azure landing zone deployments.

Considerations

- Azure Lighthouse isn't supported across Azure clouds, such as Azure public cloud to Azure Government cloud. For more information, see [Cross-region and cloud considerations](#).
- Azure Lighthouse supports delegations of subscriptions or resource groups, not management groups or tenants. For a solution to onboarding multiple subscriptions within a management group, see [Onboard all subscriptions in a management group](#). This policy follows the Azure landing zones design principle of [policy-driven governance](#).
- For information about the limitations of role support with Azure Lighthouse, see [Role support for Azure Lighthouse](#).

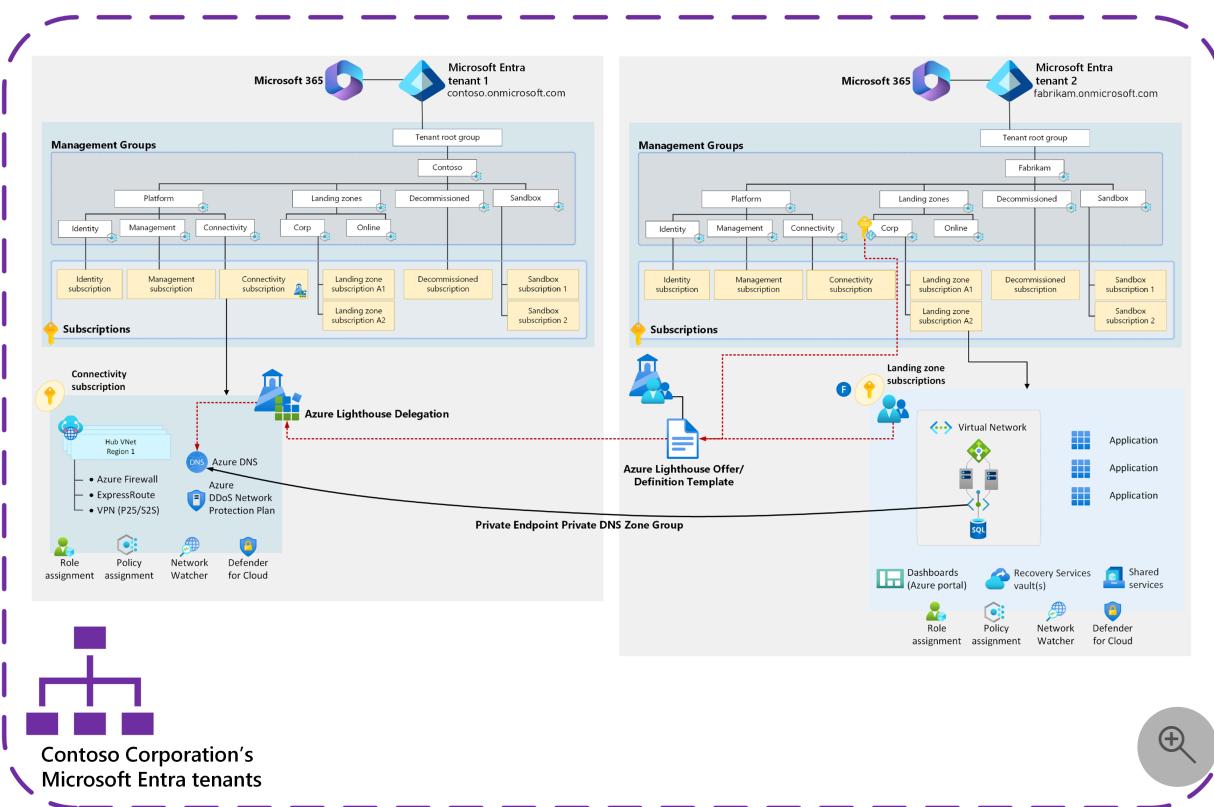
Recommendations

- See [Azure Lighthouse in enterprise scenarios](#).
- If you're an ISV, see [Azure Lighthouse in ISV scenarios](#).
- Use Azure Lighthouse in both directions between Microsoft Entra tenants to simplify management activities and reduce complex authentication and authorization scenarios. This action removes the reliance on Microsoft Entra B2B (Guest) accounts for user and workload identities, and it removes the need to have separate accounts for some activities.
- Use Microsoft Entra Privileged Identity Management (PIM) as part of your Azure Lighthouse delegations. For more information, see [Create eligible authorizations](#).
 - This feature requires Microsoft Entra ID P2 licensing but only from the source or managing Microsoft Entra tenant.

Azure landing zones scenario - Azure Lighthouse and Private DNS at scale

The following diagram is an Azure landing zone scenario where Azure Lighthouse is used across multiple Microsoft Entra tenants to assist with Private Link and DNS integration.

When you use Azure Lighthouse, Azure Policy for Private Endpoints Private DNS Zone is automatically linked in **spoke** Microsoft Entra tenants to the centralized Private DNS Zones in the **hub** Microsoft Entra tenant. For more information, see [Private Link and DNS integration at scale](#).



When you use this architecture, application landing zone owners have access to make changes to Private DNS Zone via Azure Lighthouse delegation authorizations. This access is useful if a different approach is used to manage the Private Endpoints DNS configuration, rather than Azure Policy. For more information, see [Private Link and DNS integration at scale](#).

Next steps

Considerations and recommendations for multitenant Azure landing zone scenarios

Feedback

Was this page helpful?

 Yes

 No

Identity and access management design area

Article • 11/28/2024

The identity and access management design area provides best practices that you can use to establish the foundation of your secure and fully compliant public cloud architecture.

Enterprises can have complex and heterogeneous technological landscapes, so security is critical. Robust identity and access management forms the basis of modern protection by creating a security perimeter in a public cloud. Authorization and access controls ensure that only authenticated users with verified devices can access and administer applications and resources. It ensures that the right individual can access the right resources at the right time, and for the right reason. It also provides reliable audit logging and nonrepudiation of user or workload identity actions. You should provide consistent [enterprise access control](#), including user access, control and management planes, external access, and privileged access, to improve productivity and mitigate the risk of unauthorized privilege escalation or data exfiltration.

Azure offers a comprehensive set of services, tools, and reference architectures to help your organization create highly secure and operationally efficient environments. There are several options for managing identity in a cloud environment. Each option varies in cost and complexity. Determine your cloud-based identity services based on how much you need to integrate them with your existing on-premises identity infrastructure. For more information, see [Identity decision guide](#).

Identity and access management in Azure landing zones

Identity and access management is a core consideration in both platform and application landing zones. Under the design principle of [subscription democratization](#), application owners should have the autonomy to manage their own applications and resources with minimal intervention from the platform team. Landing zones are a security boundary, and identity and access management provides a way to control the separation of one landing zone from another, along with components such as networking and Azure Policy. Apply a robust identity and access management design to help achieve application landing zone isolation.

The platform team is responsible for the foundation of identity and access management, including deploying and managing centralized directory services, such as Microsoft Entra ID, Microsoft Entra Domain Services, and Active Directory Domain Services (AD DS). Application landing zone administrators and users that access applications consume these services.

The application team is responsible for the identity and access management of their applications, including securing user access to applications and between application components, such as Azure SQL Database, virtual machines, and Azure Storage. In a well-implemented landing zone architecture, the application team can effortlessly consume services that the platform team provides.

Many of the fundamental concepts of identity and access management are the same in both platform and application landing zones, such as role-based access control (RBAC) and the principle of least privilege.

Design area review

Functions: Identity and access management requires the support of one or more of the following functions. The roles that perform these functions can help make and implement decisions.

- Cloud platform functions
- Cloud center of excellence functions
- Cloud security team functions

Scope: The goal of this design area is to help you evaluate options for your identity and access foundation. When you design your identity strategy, you should perform the following tasks:

- Authenticate users and workload identities.
- Assign access to resources.
- Determine core requirements for the separation of duties.
- Synchronize hybrid identities with Microsoft Entra ID.

Out of scope: Identity and access management forms a foundation for proper access control, but it doesn't cover more advanced aspects like:

- The Zero Trust model.
- The operational management of elevated privileges.
- Automated guardrails to prevent common identity and access mistakes.

The compliance design areas for [security](#) and [governance](#) address the out-of-scope aspects. For comprehensive recommendations for identity and access management, see [Azure identity management and access control security best practices](#).

Design area overview

Identity provides the basis for a wide variety of security assurance. It grants access based on identity authentication and authorization controls in cloud services. Access control protects data and resources and helps determine which requests should be permitted.

Identity and access management helps secure the internal and external boundaries of a public cloud environment. It's the foundation of any secure and fully compliant public cloud architecture.

The following articles examine design considerations and recommendations for identity and access management in a cloud environment:

- [Hybrid identity with Active Directory and Microsoft Entra ID](#)
- [Landing zone identity and access management](#)
- [Application identity and access management](#)

For guidance about designing solutions on Azure by using established patterns and practices, see [Identity architecture design](#).

Tip

If you have multiple Microsoft Entra ID tenants, see [Azure landing zones and multiple Microsoft Entra tenants](#).

Next steps

[Hybrid identity with Active Directory and Microsoft Entra ID](#)

Feedback

Was this page helpful?

 Yes

 No

Hybrid identity with Active Directory and Microsoft Entra ID in Azure landing zones

Article • 11/28/2024

This article provides guidance about how to design and implement Microsoft Entra ID and hybrid identity for Azure landing zones.

Organizations that operate in the cloud require a directory service to manage user identities and access to resources. Microsoft Entra ID is a cloud-based identity and access management service that provides robust capabilities to manage users and groups. You can use Microsoft Entra ID as a standalone identity solution, or integrate it with a Microsoft Entra Domain Services infrastructure or an on-premises Active Directory Domain Services (AD DS) infrastructure.

Microsoft Entra ID provides modern secure identity and access management for Azure and Microsoft 365 services. You can use Microsoft Entra ID for various organizations and workloads. For example, organizations with an on-premises AD DS infrastructure and cloud-based workloads can use directory synchronization with Microsoft Entra ID. Directory synchronization ensures that on-premises and cloud environments share a consistent set of identities, groups, and roles. Applications that require legacy authentication mechanisms might need Domain Services for managed domain services in the cloud and to extend the on-premises AD DS infrastructure into Azure.

Cloud-based identity management is an iterative process. You might start with a cloud-native solution that has a small set of users and corresponding roles for an initial deployment. As your migration matures, you might need to integrate your identity solution by using directory synchronization or add cloud-hosted domain services as part of your cloud deployments.

Adjust your identity solution over time depending on your workload authentication requirements and other needs, such as changes to your organizational identity strategy and security requirements or integration with other directory services. When you evaluate Windows Server Active Directory solutions, understand the differences between Microsoft Entra ID, Domain Services, and AD DS on Windows Server.

For more information, see [Identity decision guide](#).

Identity and access management services in Azure landing zones

The platform team is responsible for the administration of identity and access management. Identity and access management services are fundamental to organizational security. Your organization can use Microsoft Entra ID to control administrative access and protect platform resources. This approach prevents users outside of the platform team from making changes to the configuration or to the security principals contained within Microsoft Entra ID.

If you use AD DS or Domain Services, you must protect the domain controllers from unauthorized access. Domain controllers are highly vulnerable to attacks and should have strict security controls and segregation from application workloads.

Deploy domain controllers and associated components, such as Microsoft Entra Connect servers, in the Identity subscription that's located in the platform management group. Domain controllers aren't delegated to application teams. This isolation allows application owners to use identity services without having to maintain them, which reduces the risk of compromise to identity and access management services. The resources in the Identity platform subscription are a critical point of security for your cloud and on-premises environments.

Provision landing zones so that application owners can choose either Microsoft Entra ID or AD DS and Domain Services to suit their workload needs. You might need to configure other services, depending on your identity solution. For example, you might need to enable and secure network connectivity to the Identity virtual network. If you use a subscription-vending process, include this configuration information in your subscription request.

Azure and on-premises domains (hybrid identity)

User objects that you create entirely in Microsoft Entra ID are known as *cloud-only accounts*. Cloud-only accounts support modern authentication and access to Azure and Microsoft 365 resources and support access for local devices that use Windows 10 or Windows 11.

Your organization might already have longstanding AD DS directories that you integrate with other systems, such as line of business or enterprise resource planning (ERP) through the Lightweight Directory Access Protocol (LDAP). These domains can have many domain-joined computers and applications that use Kerberos or older NTLMv2

protocols for authentication. In these environments, you can synchronize user objects to Microsoft Entra ID so that users can sign in to both on-premises systems and cloud resources with a single identity. Unifying on-premises and cloud directory services is known as *hybrid identity*. You can extend on-premises domains into Azure landing zones:

- To maintain a single user object in both cloud and on-premises environments, you can sync AD DS domain users with Microsoft Entra ID through Microsoft Entra Connect or Microsoft Entra Cloud Sync. To determine the recommended configuration for your environment, see [Topologies for Microsoft Entra Connect](#) and [Topologies for Microsoft Entra Cloud Sync](#).
- To domain join Windows virtual machines (VMs) and other services, you can deploy AD DS domain controllers or Domain Services in Azure. Use this approach so that AD DS users can sign in to Windows servers, Azure file shares, and other resources that use Active Directory as an authentication source. You can also use other Active Directory technologies, like group policy, as an authentication source. For more information, see [Common deployment scenarios for Microsoft Entra Domain Services](#).

Hybrid identity recommendations

- To determine your identity solution requirements, document the authentication provider that each application uses. Use the [identity decision guide](#) to select the right services for your organization. For more information, see [Compare Active Directory to Microsoft Entra ID](#).
- You might use [Domain Services](#) for applications that rely on domain services and use older protocols. Existing AD DS domains sometimes support backward compatibility and allow legacy protocols, which can negatively affect security. Instead of extending an on-premises domain, consider using Domain Services to create a new domain that doesn't allow legacy protocols. Use the new domain as the directory service for cloud-hosted applications.
- Factor in resiliency as a critical design requirement for your hybrid identity strategy in Azure. Microsoft Entra ID is a [globally redundant, cloud-based system](#) but Domain Services and AD DS aren't. Carefully plan for resiliency when you implement Domain Services and AD DS. When you design either service, consider using multiregion deployments to ensure continued service operation in the event of a regional incident.

- To extend an on-premises AD DS instance into Azure and optimize deployment, incorporate your Azure regions into your Active Directory [site design](#). Create a site in AD DS sites and services for each Azure region where you plan to deploy workloads. Then create a new subnet object in AD DS sites and services for each IP address range that you plan to deploy in the region. Associate the new subnet object with the AD DS site that you created. This configuration ensures that the domain controller locator service directs authorization and authentication requests to the nearest AD DS domain controllers within the same Azure region.
- Evaluate scenarios that set up guests, customers, or partners so that they can access resources. Determine whether these scenarios involve [Microsoft Entra B2B](#) or [Microsoft Entra External ID for customers](#). For more information, see [Microsoft Entra External ID](#).
- Don't use [Microsoft Entra application proxy](#) for intranet access because it adds latency to the user experience. For more information, see [Microsoft Entra application proxy planning](#) and [Microsoft Entra application proxy security considerations](#).
- Consider various methods that you can use to [integrate on-premises Active Directory with Azure](#) to meet your organizational requirements.
- If you have Active Directory Federation Services (AD FS) federation with Microsoft Entra ID, you can use password hash synchronization as a backup. AD FS doesn't support Microsoft Entra seamless single sign-on (SSO).
- Determine the right [synchronization tool](#) for your cloud identity.
- If you have requirements for using AD FS, see [Deploy AD FS in Azure](#).
- If you use [Microsoft Entra Connect](#) as your synchronization tool, consider deploying a [staging server](#) in a region that's different from your primary Microsoft Entra Connect server for disaster recovery. Alternatively, if you don't use multiple regions, implement availability zones for high availability.
- If you use [Microsoft Entra Cloud Sync](#) as your synchronization tool, consider installing at least [three agents](#) across different servers in multiple regions for disaster recovery. Alternatively, you can install the agents across servers in different availability zones for high availability.

 **Important**

We highly recommend that you migrate to Microsoft Entra ID unless you specifically require AD FS. For more information, see [Resources for](#)

[decommissioning AD FS](#) and [Migrate from AD FS to Microsoft Entra ID](#).

Microsoft Entra ID, Domain Services, and AD DS

To implement Microsoft directory services, familiarize administrators with the following options:

- You can [deploy AD DS domain controllers into Azure as Windows VMs](#) that platform or identity administrators fully control. This approach is an infrastructure as a service (IaaS) solution. You can join the domain controllers to an existing Active Directory domain or host a new domain that has an optional trust relationship with existing on-premises domains. For more information, see [Azure Virtual Machines baseline architecture in an Azure landing zone](#).
- [Domain Services](#) is an Azure-managed service that you can use to create a new managed Active Directory domain that's hosted in Azure. The domain can have a trust relationship with existing domains and can synchronize identities from Microsoft Entra ID. Administrators don't have direct access to the domain controllers and aren't responsible for patching and other maintenance operations.
- When you deploy Domain Services or integrate on-premises environments into Azure, use [regions with availability zones](#) for increased availability when possible. Also consider deploying across multiple Azure regions for increased resiliency.

After you configure AD DS or Domain Services, you can use the same method as on-premises computers to domain join Azure VMs and file shares. For more information, see [Compare Microsoft directory-based services](#).

Microsoft Entra ID and AD DS recommendations

- Use [Microsoft Entra application proxy](#) to access applications that use on-premises authentication remotely through Microsoft Entra ID. This feature provides secure remote access to on-premises web applications. Microsoft Entra application proxy doesn't require a VPN or any changes to the network infrastructure. However, it's deployed as a single instance into Microsoft Entra ID, so application owners and the platform or identity teams must collaborate to ensure that the application is configured correctly.
- Evaluate the compatibility of workloads for AD DS on Windows Server and Domain Services. For more information, see [Common use cases and scenarios](#).

- Deploy domain controller VMs or Domain Services replica sets into the Identity platform subscription within the platform management group.
- Secure the virtual network that contains the domain controllers. To prevent direct internet connectivity to and from the virtual network and domain controller, place the AD DS servers in an isolated subnet with a network security group (NSG). The NSG provides firewall functionality. Resources that use domain controllers for authentication must have a network route to the domain controller subnet. Enable a network route only for applications that require access to services in the Identity subscription. For more information, see [Deploy AD DS in an Azure virtual network](#).
- Use [Azure Virtual Network Manager](#) to enforce standard rules that apply to virtual networks. For example, you can use Azure Policy or virtual network resource tags to add landing zone virtual networks to a network group if they require Active Directory identity services. The network group can then be used that allows access to the domain controller subnet only from applications that require it and block the access from other applications.
- Secure the role-based access control (RBAC) permissions that apply to the domain controller VMs and other identity resources. Administrators with RBAC role assignments at the Azure control plane, such as Contributor, Owner, or Virtual Machine Contributor, can run commands on the VMs. Ensure that only authorized administrators can access the VMs in the Identity subscription and that overly permissive role assignments aren't inherited from higher management groups.
- Keep your core applications close to, or in the same region as, the virtual network for your replica sets to minimize latency. In a multiregional organization, deploy Domain Services into the region that hosts the core platform components. You can only deploy Domain Services into a single subscription. To expand Domain Services to further regions, you can add up to four more [replica sets](#) in separate virtual networks that are peered to the primary virtual network.
- Consider deploying AD DS domain controllers into multiple Azure regions and across [availability zones](#) to increase resiliency and availability. For more information, see [Create VMs in availability zones](#) and [Migrate VMs to availability zones](#).
- Explore the [authentication methods for Microsoft Entra ID](#) as part of your identity planning. Authentication can occur in the cloud and on-premises, or on-premises only.
- Consider using [Kerberos authentication for Microsoft Entra ID](#) instead of deploying domain controllers in the cloud if a Windows user requires Kerberos for Azure Files

file shares.

Next step

Landing zone identity and access management

Feedback

Was this page helpful?

 Yes

 No

Landing zone identity and access management

Article • 08/14/2024

After you identify your identity architecture, you need to manage the authorization and access for resources in application and platform landing zones. Consider which resources each authenticated principal has access to and needs access to, and how to mitigate the risk of unauthorized access to your resources. For more information, see [Identity architecture design](#).

Overview

The identity and access management design area provides guidance to help you implement the [enterprise access model in Azure](#) and implement and secure control planes. When you incorporate the design principle of [subscription democratization](#), your application team can manage their own workloads within the policy guardrails that the platform team sets. This approach also follows the [policy-driven governance](#) principle.

The platform team is responsible for provisioning new application landing zones or subscriptions. When they provision a landing zone for an application owner, the platform team should configure it with the appropriate access controls so the application owner can manage their own resources. The application owner should be able to create and manage users and groups within Microsoft Entra ID, and assign roles to those users and groups. The application owner can then manage access to their own resources and delegate access to other users and groups as required. The landing zone should also have optional network connectivity to Active Directory Domain Services (AD DS) or Microsoft Entra Domain Services in the Microsoft identity platform subscription, depending on the requirements of the application.

Use Azure role-based access control (RBAC) to manage administrative access to Azure resources. Consider whether users require permissions on a narrow scope, such as an Administrator for a single application, or a broad scope, such as a Network Administrator across multiple application workloads. In either case, follow the principle of just-enough access, and ensure that the user has only the roles required for their normal activities. Use custom roles and Microsoft Entra Privileged Identity Management (PIM) where necessary to enforce just-in-time (JIT) access. Although the platform team is responsible for the identity and access management foundation, both platform and application teams are consumers of the service and should follow the same principles.

Identity and access management is important for the successful separation of one landing zone from another and the isolation of workloads within an organization. It's a critical design area for both platform and application landing zones.

If your organization uses a [subscription-vending process](#), you can automate many of the identity and access configurations for application landing zones. Implement subscription vending to help standardize landing zone creation and so application teams can manage their own resources.

Design considerations

Some organizations share services between multiple applications. For example, there might be a centralized integration service used by several independent applications. In that scenario, consider which services are managed centrally and which are devolved to application teams, and understand where security boundaries need to be enforced. Giving application teams administrative access to the shared service might be helpful for developer productivity, but might provide more access than is required.

Managing application resources that don't cross security boundaries can be delegated to application teams. Consider delegating other aspects that are required to maintain security and compliance as well. Letting users provision resources within a securely managed environment lets organizations take advantage of the agile nature of the cloud and prevent violation of any critical security or governance boundary.

RBAC

Important

Classic resources and classic administrators are [retiring on August 31, 2024](#). Remove unnecessary co-administrators, and use Azure RBAC for fine-grained access control.

Understand the difference between Microsoft Entra ID roles and Azure RBAC roles.

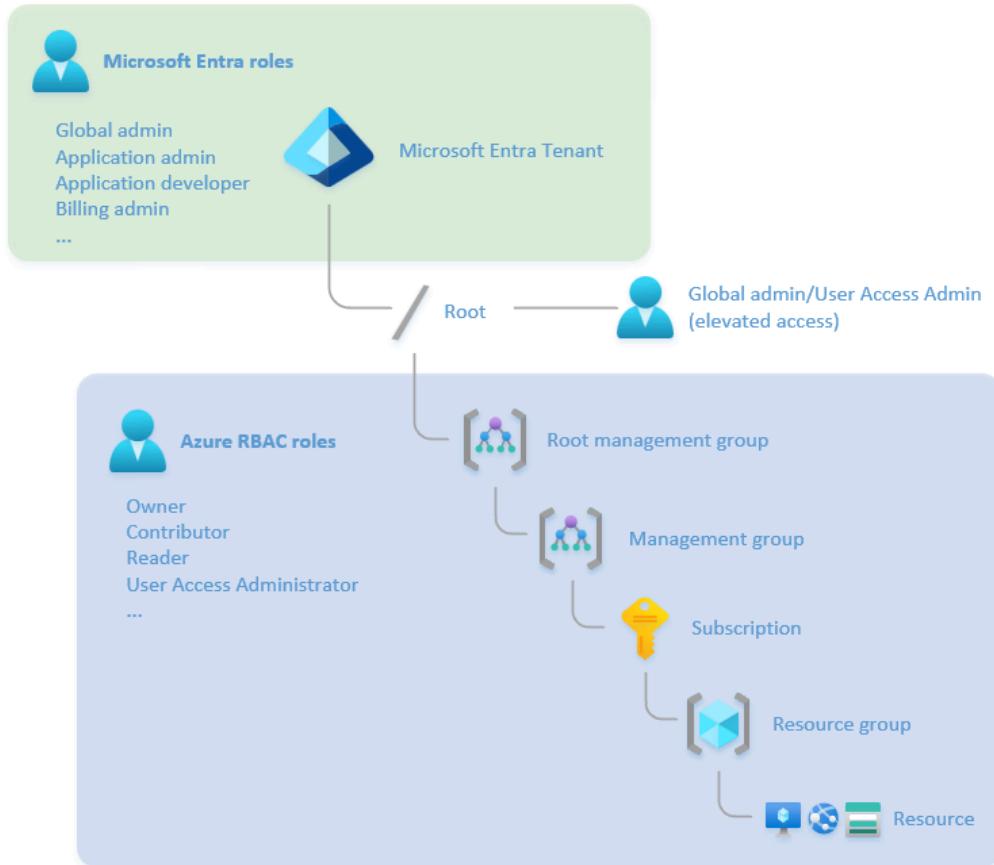
- [Microsoft Entra ID roles](#) control the administrative privileges to tenant-wide services such as Microsoft Entra ID, and other Microsoft services including Microsoft Teams, Microsoft Exchange Online, and Microsoft Intune.

- Azure RBAC roles control the administrative privileges to Azure resources such as virtual machines, subscriptions, and resource groups.
- The Azure RBAC Owner and User Access Administrator roles can modify the role assignments on Azure resources. By default, the Microsoft Entra Global Administrator role doesn't have permission to manage access to Azure resources. It must be explicitly enabled. For more information, see [Elevate access to manage all Azure subscriptions and management groups](#).

i Important

Microsoft recommends that you use roles with the fewest permissions. This helps improve security for your organization. Global Administrator is a highly privileged role that should be limited to emergency scenarios when you can't use an existing role.

The following diagram shows the relationship between Microsoft Entra ID roles and Azure RBAC roles:



- You can create role-assignable groups and [assign Microsoft Entra roles to the groups](#) if you set the `isAssignableToRole` property to `true`. Only groups with this property set are protected. The only roles that can modify a group's membership are Global Administrators, Privileged Role Administrators, or the group's owner.
- Only [some roles can reset the password](#) or multifactor authentication (MFA) settings for another administrator. This restriction prevents unauthorized administrators from resetting the credentials of a higher-privileged account to get more permissions.
- If the Azure built-in roles don't meet the specific needs of your organization, you can [create your own custom roles](#). Just like built-in roles, you can assign custom roles to users, groups, and service principals at tenant, management group, subscription, and resource group scopes. Aim to use Azure built-in roles where possible, and only create custom roles when necessary.
- When you design your access control strategy, know the [service limits for roles](#), role assignments, and custom roles.
- Some Azure RBAC roles support [attribute-based access control \(ABAC\)](#), or role assignment conditions. When you use conditions, administrators can dynamically assign roles based on the attributes of the resource. For example, you can assign the Storage Blob Data Contributor role but only for blobs that have a specific index tag rather than all blobs in a container.
- You can use built-in and custom RBAC roles with `Microsoft.Authorization/roleAssignments/write` or `Microsoft.Authorization/roleAssignments/delete` permissions to create, delete, and update role assignments. Anyone that has this role can decide who has write, read, and delete permissions for any resource in the assignment scope. Platform or application

landing zone team members should consider how to delegate privileged roles to other users and groups to grant them necessary autonomy. To ensure compliance with least-privilege access principles, they can use [conditions](#) to delegate users.

Design recommendations

General recommendations

- Enforce [Microsoft Entra multifactor authentication \(MFA\)](#) for users that have rights to the Azure environment, including the platform subscription, the application subscription, and the Microsoft Entra ID tenant. Many compliance frameworks require MFA enforcement. MFA helps to reduce the risk of credential theft and unauthorized access. To prevent unauthorized access to sensitive information, ensure that you include users with Reader roles in MFA policies.
- Use [Microsoft Entra Conditional Access](#) policies for users that have rights to the Azure environment. Conditional Access is another feature that helps protect a controlled Azure environment from unauthorized access. Application and platform administrators should have Conditional Access policies that reflect the risk profile of their role. For example, you might have requirements to carry out administrative activities only from specific locations or specific workstations. Or the sign-in risk tolerance for users with administrative access to Azure resources might be lower than it is for standard Microsoft Entra ID users.
- Enable [Microsoft Defender for Identity](#) to help protect user identities and secure user credentials. Defender for Identity is part of Microsoft Defender XDR. You can use Defender for Identity to identify suspicious user activities and get incident timelines. You can also use it with Conditional Access to deny high-risk authentication attempts. Deploy Defender for Identity sensors onto on-premises domain controllers and domain controllers in the Azure identity subscription.
- Use [Microsoft Sentinel](#) to provide threat intelligence and investigative capabilities. Sentinel uses logs from Azure Monitor Logs, Microsoft Entra ID, Microsoft 365, and other services to provide proactive threat detection, investigation, and response.
- Separate administrative access from nonadministrative, day-to-day access, such as web browsing and email access. Web and email are common attack vectors. When a user account is compromised, it's less likely to result in a security breach if the account isn't used for administrative access.
 - Use [separate, cloud-only accounts for privileged roles](#). Don't use the same account for daily use that you do for privileged administration. Privileged Microsoft Entra ID and Azure RBAC roles are marked as *PRIVILEGED* in the Azure portal and in documentation.
 - For nonprivileged job function roles that can manage Azure application resources, consider whether you require separate administrative accounts or use [Microsoft Entra PIM](#) to control administrative access. PIM ensures that the account has the required permissions only when needed and that the permissions are removed when the task is complete (also known as *just-in-time access*).
- To make role assignments more manageable, don't assign roles directly to users. Instead, assign roles to groups to help minimize the number of role assignments, which has a [limit for each subscription](#).
 - Use [Microsoft Entra PIM for groups](#) to apply just-in-time administrative access controls to privileged users. Consider controlling group membership with [entitlement management](#). You can use the entitlement management feature to add approval and auditing workflows to group membership operations and help ensure that administrative group members aren't unnecessarily added or removed.
 - When you grant access to resources, use Microsoft Entra-only groups for Azure control-plane resources. Both Entra-only users and groups, and those synchronized from on-premises using Microsoft Entra Connect, can be added to an Entra-only group. Add on-premises groups to the Microsoft Entra-only group if a group management system is already in place. Using Entra-only groups helps protect the cloud control plane from unauthorized modification of on-premises directory services. Note that *Microsoft Entra-only* is also known as *cloud only*.
- Create [emergency-access](#) accounts, or break-glass accounts, to avoid accidentally being locked out of your Microsoft Entra ID organization. Emergency-access accounts are highly privileged and are only assigned to specific individuals. Store the credentials for the accounts securely, monitor their use, and test them regularly to ensure that you can use them if there's a disaster.

For more information, see [Secure access practices for administrators in Microsoft Entra ID](#).

Microsoft Entra ID recommendations

- Integrate [Microsoft Entra ID with Azure Monitor](#) so you can analyze your sign-in activity and the audit trail of changes within your tenant. Configure a diagnostic setting to send sign-in logs and audit logs to the platform central Azure Monitor Logs workspace in

the management subscription.

- Use the entitlement management feature of Microsoft Entra ID Governance to [create access packages](#) that control group membership via automatic approval processes and regular access reviews for privileged group members.
- Use [Microsoft Entra built-in roles](#) to manage the following identity settings from a tenant level:

 Expand table

Role	Description	Note
Global Administrator	Manages all aspects of Microsoft Entra ID and Microsoft services that use Microsoft Entra identities.	Don't assign more than five people to this role.
Hybrid Identity Administrator	Manages cloud provisioning from Active Directory to Microsoft Entra ID and also manages Microsoft Entra Connect, Microsoft Entra pass-through authentication, Microsoft Entra password hash synchronization, Microsoft Entra seamless single sign-on (SSO), and federation settings.	
Security Administrator	Reads security information and reports, and manages configurations in Microsoft Entra ID and Microsoft 365.	
Application Administrator	Creates and manages all aspects of app registrations and enterprise apps.	You can't grant tenant-wide administrative consent.

- Don't assign a higher-privileged role to a task that a lower-privileged role can do. For example, assign the User Administrator role to manage users, not the Global Administrator role. For more information, see [Microsoft Entra built-in roles permissions](#).
- Use [administrative units](#) to restrict a set of administrators so they can only manage specific objects in your tenant. You can use administrative units to delegate the administration of a subset of the directory. For example, you can delegate the administration of a service desk to a single business unit within a wider organization.

Administrative units can also help eliminate the need for separate Microsoft Entra ID tenants as a security boundary, where separate teams manage the Microsoft 365 platform and the Azure platform in the same organization. For example, you can use administrative units to delegate the management of Azure application security principals to the application team without granting privileges on the entire Microsoft Entra ID tenant.

- Use [restricted management administrative units](#) to provide further protection. Prevent anyone other than a specific set of administrators that you designate from modifying specific objects. For example, your separation of duty policies might require that you use this feature to prevent anyone from modifying a specific user account, even users with the User Administrator role. This restriction is useful for service accounts that applications use and that even administrators shouldn't modify. You can also prevent privilege escalation, for example if someone modifies a user account or group that has platform or landing zone administration privileges.

Azure RBAC recommendations

- To simplify administration and reduce the risk of misconfiguration, standardize roles and role assignments across all application landing zones. For example, if you have a role that delegates users to manage virtual machines, use the same role in all application landing zones. This approach also simplifies the process of moving resources between landing zones.
- Use [Azure RBAC](#) to manage data plane access to resources, if possible. Examples of data plane endpoints are Azure Key Vault, a storage account, or a SQL database.
- Ensure that Azure Monitor Logs workspaces are configured with the appropriate permission model. When you use a centralized Azure Monitor Logs workspace, use [resource permissions](#) to ensure that application teams have access to their own logs but not to logs from other teams.

Built-in roles

- Consider whether [built-in roles](#) are suitable for your requirements. In many cases, you can assign multiple built-in roles to a security group to provide the appropriate access for a user. But sometimes, you can't use built-in roles and also comply with least-privilege access because the roles might include permissions that exceed what your users require. For more granular control, consider creating a custom role that reflects the specific permissions required to carry out a job function. For more information, see [Provide role-based authorization](#).

- Many Azure built-in roles provide predefined role assignments at the platform and resource level. When you [combine several role assignments](#), consider the overall effects.
- The Azure landing zone accelerator includes several custom roles for common administrative functions. You can use these roles alongside Azure built-in roles. The following table describes the custom administrative roles or areas for the Azure landing zone accelerator:

[Expand table](#)

Administrative role or area	Description	Actions	NotActions
Azure Platform Owner (such as the built-in Owner role)	Manages management groups and subscription lifecycles	*	
Subscription Owner	Delegated role for the subscription owner	*	<code>Microsoft.Authorization/*/write,</code> <code>Microsoft.Network/vpnGateways/*,</code> <code>Microsoft.Network/expressRouteCircuits/*,</code> <code>Microsoft.Network/routeTables/write,</code> <code>Microsoft.Network/vpnSites/*</code>
Application Owner (DevOps, App operations)	Contributor role for the application or operations team at the subscription scope	*	<code>Microsoft.Authorization/*/write,</code> <code>Microsoft.Network/publicIPAddresses/write,</code> <code>Microsoft.Network/virtualNetworks/write,</code> <code>Microsoft.KeyVault/locations/deletedVaults/purge/action</code>
Network management (NetOps)	Manages platform-wide global connectivity, such as virtual networks, UDRs, NSGs, NVAs, VPNs, Azure ExpressRoute, and others	<code>*/read,</code> <code>Microsoft.Network/*,</code> <code>Microsoft.Resources/deployments/*,</code> <code>Microsoft.Support/*</code>	
Security operations (SecOps)	Security Administrator role with a horizontal view across the entire Azure estate and the Key Vault purge policy	<code>*/read,</code> <code>*/register/action,</code> <code>Microsoft.KeyVault/locations/deletedVaults/purge/action,</code> <code>Microsoft.PolicyInsights/*,</code> <code>Microsoft.Authorization/policyAssignments/*,</code> <code>Microsoft.Authorization/policyDefinitions/*,</code> <code>Microsoft.Authorization/policyExemptions/*,</code> <code>Microsoft.Authorization/policySetDefinitions/*,</code> <code>Microsoft.Insights/alertRules/*,</code> <code>Microsoft.Resources/deployments/*,</code> <code>Microsoft.Security/*,</code> <code>Microsoft.Support/*</code>	

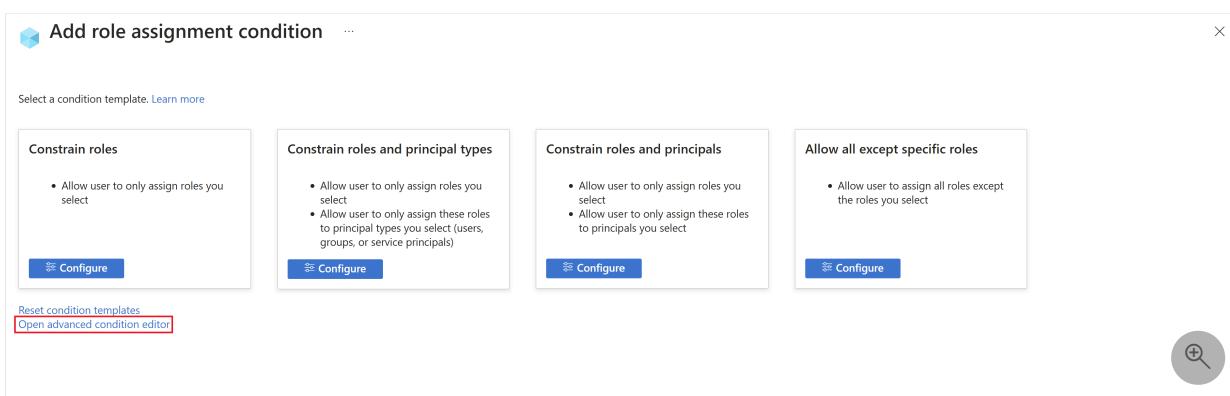
Those roles might need extra rights depending on the responsibility model. For example, in some organizations a NetOps role might only need to manage and configure global connectivity. In organizations that need a more centralized approach, you can enrich the NetOps role with more allowed actions, such as creating peering between hubs and their spokes.

Role assignments and groups

- When the platform team provisions an application landing zone, they should ensure that all required identity and access management objects are created, such as security groups, standard role assignments, and user-assigned managed identities.
- Create landing zone role assignments at the subscription or resource group scope. Azure Policy assignments occur at the management group scope, so you should provision landing zone role assignments at a lower scope. Use this approach to ensure

that landing zone administrators have full autonomy over their resources but can't modify the Azure Policy assignments that govern their landing zone.

- Each application landing zone should have its own groups and role assignments. Don't create generic groups and assign them to multiple landing zones. This approach can lead to misconfiguration and security breaches, and it's difficult to manage at scale. If one user requires access to multiple landing zones, assign them to the appropriate groups in each landing zone. Use ID Governance to manage their group membership.
- Assign roles to groups, not to users. This approach helps to ensure that users have the correct permissions when they join or leave your organization. It also helps to ensure that users have the correct permissions when they move between teams. For example, if a user moves from the network team to the security team, you should remove them from the network group and add them to the security group. If you assign a role directly to a user, they retain the role after moving to a different team. Use ID Governance to manage group membership rather than manually adding and removing group members.
- Maintain separate security configurations for different environments of the same application, such as dev/test and production. Create separate groups and role assignments for each environment. Don't share managed identities or service principals across environments. Treat each environment as a separate landing zone. This approach helps to ensure isolation between dev/test and production, and standardizes the process of moving application deployments between environments. If the same individual requires access to several landing zones, you should assign them to the appropriate groups in each landing zone.
- Consider whether platform administrators require permissions on application landing zones. If so, use Microsoft Entra PIM to control access to those resources, and assign the least-privileged permissions required. For example, a platform administrator might require access to a specific application landing zone to troubleshoot an issue but shouldn't have routine access to the application data or code. In this case, the platform administrator can request access to the application. A privileged role administrator approves the request, and the platform administrator is granted the required permissions for the specified time period. This approach helps enforce separation of duties and protects application landing zones from accidental or malicious misconfiguration.
- When you delegate administrative responsibility to others, such as application teams, consider whether they require the full set of privileges or only a subset. Follow the principle of least privilege (PoLP). For example, you might assign the User Access Administrator role or RBAC Administrator role to a user who needs to manage access to Azure resources but doesn't need to manage the resources themselves. To limit the identities, identity types, and roles that users can delegate and assign Azure RBAC assignments to, use [delegated role assignments with conditions](#). Application teams can use conditions to manage their own security principals within the constraints that the platform team sets. More privileged role assignments require escalation to the platform team. Consider the following factors when you use conditions to delegate RBAC roles:
 - Review current role assignments for built-in and custom privileged roles and evaluate if you should add appropriate conditions to those existing assignments. For example, you can add conditions to the Subscription Owner and Application Owner custom roles that the Azure landing zone accelerator provides. These conditions can restrict the principal types that they can assign roles to or limit specific roles that they can assign.
 - Follow the PoLP when you add conditions to role assignments. For example, limit delegates to only assign roles to groups or enable delegates to assign all roles except privileged administrator roles like Owner, User Access Administrator, and RBAC Administrator.
 - Build your own conditions if the available condition templates don't fulfill your requirements or policies.



- Review the [known limitations](#) of delegating Azure access management to others.
- The following table shows an example role assignment structure for an Azure landing zone environment. It provides a balance between security and ease of administration. You can adapt the structure to suit your organization's requirements. You can assign the same individual to multiple groups, depending on their role within the organization. But you should apply the RBAC assignments to a specific group within a specific landing zone.

Resource	User	Role assignment	Assignment target	Assignment scope
Application X landing zone	Application X owners	Application Owner (custom, included in Azure landing zone accelerator)	Application X Admins security group	Application X production and dev/test subscriptions
Application X landing zone	Application X owners	Application Access Administrator (custom, with role assignment conditions to manage access to their own application)	Application X Admins security group	Application X production and dev/test subscriptions
Application X landing zone	Application X data administrator	Data Administrator (custom, with permissions on required data resources)	Application X Data Team security group	Application X production and dev/test subscriptions
Application Y landing zone	Application Y owners	Application Owner (custom, included in Azure landing zone accelerator)	Application Y Admins security group	Application Y production and dev/test subscriptions
Application Y landing zone	Application Y testing team	Test Contributor (custom, with permissions required for application testing)	Application Y Test Team security group	Application Y dev/test subscription
Sandbox	Application Z development team	Owner (built-in)	Application Z developers security group	Application Z resource groups in sandbox subscription
Platform resources	Platform management team	Contributor (built-in)	Platform Admins PIM group	Platform management group
Platform landing zones	Platform management team	Reader (built-in)	Platform Team security group	Organizational top-level management group
Tenant-wide	Security team	Security Operations (custom, included in Azure landing zone accelerator)	Security Ops security group	Organizational top-level management group
Tenant-wide	Security team	Conditional Access Administrator (built-in, with protected actions enabled)	Security administrators security group	Microsoft Entra ID tenant
Tenant-wide	Network team	Network Operations (Custom, included in Azure landing zone accelerator)	Network Ops security group	All subscriptions
Tenant-wide	FinOps team	Billing Reader (built-in)	FinOps Team security group	Organizational top-level management group

- Azure Policy assignments that have the `DeployIfNotExists` effect require a [managed identity](#) to remediate noncompliant resources. If you use a system-assigned managed identity as part of the Azure Policy assignment process, Azure automatically grants the required permissions. If you use a user-assigned managed identity, the permissions must be granted manually. The managed identity role assignments must follow the PoLP and enable only the required permissions to carry out the policy remediation on the target scope. Policy remediation managed identities don't support custom role definitions. Apply role assignments directly to managed identities and not to groups.

Microsoft Entra PIM recommendations

- Use [Microsoft Entra PIM](#) to comply with the Zero Trust model and least-privilege access. Correlate your organization's roles to the minimum access levels needed. In Microsoft Entra PIM, you can use Azure-native tools, extend existing tools and processes, or use both existing and native tools as needed.
- Use [Microsoft Entra PIM access reviews](#) to regularly validate resource entitlements. Access reviews are part of many compliance frameworks, so many organizations already have an access review process in place.
- Use privileged identities for automation runbooks that require elevated access permissions, or for privileged deployment pipelines. You can use the same tools and policies to govern automated workflows that access critical security boundaries that you use to govern users of equivalent privilege. Automation and deployment pipelines for application teams should have role assignments that prevent an application owner from escalating their own privileges.
- Control highly privileged Azure RBAC roles, such as Owner or User Access Administrators that are assigned to platform or application landing zone team members on a subscription or management group. Use [Microsoft Entra PIM for groups](#) to configure Azure RBAC roles so they require the same elevation process as Microsoft Entra ID roles.

For example, a user might routinely require limited administrative access to resources in an application landing zone. Occasionally, they might require the Owner role. You can create two security groups: Application Administrators and Application Owners. Assign

the least-privilege roles to the Application Administrators group, and assign the owner role to the Application Owners role. Use PIM groups so the user can request the Owner role when required. At all other times, the user has only the permissions required to carry out their typical activities.

- Use [protected actions](#) with Microsoft Entra PIM to add extra layers of protection. In Microsoft Entra ID, protected actions are permissions that are assigned [Conditional Access policies](#). When a user attempts to perform a protected action, they must first satisfy the Conditional Access policies that are assigned to the required permissions. For example, to allow administrators to update cross-tenant access settings, you can require that they first satisfy the [phishing-resistant MFA policy](#).

Identity and access management in the Azure landing zone accelerator

Identity and access management are core features of Azure landing zone accelerator implementation. The deployment includes a subscription that's dedicated to identity, where organizations can deploy AD DS domain controllers or other identity services, such as Microsoft Entra Connect servers, that are required for their environment. Not all organizations require services in the subscription. For example, some organizations might have applications that are already fully integrated with Microsoft Entra ID.

The identity subscription has a virtual network that's peered to the hub virtual network in the platform subscription. With this configuration, the platform team can manage the identity subscription, and application owners have access to identity services as required. You must secure the identity subscription and virtual network to protect identity services from unauthorized access.

Azure landing zone accelerator implementation also includes options to:

- Assign recommended policies to govern identity and domain controllers.
- Create a virtual network, and connect to the hub via virtual network peering.

Next steps

[Application identity and access management](#)

Feedback

Was this page helpful?



Application identity and access management

Article • 11/28/2024

This article describes considerations and recommendations that application owners and developers can use to design the identity and access management for cloud-native applications.

If your team migrates or creates cloud-native applications, you must consider the authentication and access requirements for the applications. These requirements determine how users authenticate to applications and how application resources authenticate to each other, for example when a web application accesses a SQL database.

In the [platform automation and DevOps design area](#), we recommend that your application team transitions workloads to [subscription vending](#). As part of the subscription-vending process, your application team needs to provide identity and access requirements to the platform team so they can create the appropriate subscriptions. Application owners are responsible for the identity and access management of individual applications. They should manage their application by using the centralized services that the platform team provides.

Design considerations

To help reduce the risk of unauthorized access to your applications, incorporate the following considerations into your design.

- There are several authentication and authorization standards, like OAuth 2.0, OpenID Connect, JSON web tokens (JWTs), and SAML (Security Assertion Markup Language). Determine which [authentication and authorization standards](#) to use for your application.
- When you request an application landing zone from the platform team, you can help ensure that they create the appropriate subscriptions by asking them the following questions:
 - How will end users authenticate to and access the application?
 - Who needs role-based access control (RBAC) permissions for resources and services that the application uses?

- Do existing built-in roles cover the RBAC access requirements for both control plane and data plane access, or do you need to create new custom roles?
- Did the platform team implement any compliance policies that might cause problems with the application?
- Which application components need to communicate with each other?
- Are there any requirements for accessing the shared resources, such as Microsoft Entra Domain Services, that are deployed in the platform landing zone?

Azure Key Vault and managed identities

- Security breaches of public cloud resources often originate from leaked credentials that are embedded in code or other text. You can use managed identities and [Key Vault](#) to implement programmatic access and help reduce the risk of credential theft.
- If your application or workload requires a service to securely store credentials, you can use Key Vault to manage secrets, keys, and certificates.
- To avoid having credentials in your code, you can use managed identities with Azure VMs to authenticate to any service that [supports Microsoft Entra ID authentication](#). For more information, see [Use managed identities for Azure resources on a VM to acquire an access token](#).
- [Managed identities](#) provide an automatically managed identity principal that applications and resources use when they connect to resources that support Microsoft Entra ID authentication. Applications can use managed identities to [obtain Microsoft Entra ID tokens without having to manage any credentials](#).
 - You can use [system-assigned or user-assigned managed identities](#).
 - It's easy to confuse how service principals and managed identities access Azure resources. To understand the difference between the two, see [Demystifying service principals—Managed identities ↗](#).
 - Where possible, use managed identities to support authentication rather than using service principals and Microsoft Entra ID app registrations. You must have the Application Administrator or Application Developer RBAC roles to create service principals and app registrations. These privileged roles are typically assigned to the platform team or identity team. Use managed identities to

eliminate the need for the platform team to create service principals and app registrations for your application team.

- You can use managed identities to authenticate to any service that supports Microsoft Entra authentication. However, not all services support managed identities to access other services. For some services, it might be necessary to store credentials. You should securely store credentials, avoid sharing credentials with other services, and follow the principle of least privilege. For more information, see [Azure services that can use managed identities to access other services](#).
- You can use managed identities with Azure virtual machines (VMs) to authenticate to any service that [supports Microsoft Entra ID authentication](#). For more information, see [Use managed identities for Azure resources on a VM to acquire an access token](#).
- There are restrictions on moving resources with managed identities between subscriptions and regions. For example, you might move resources between subscriptions or regions for a merger, acquisition, or repatriation of resources for data sovereignty reasons.

If an Azure resource has user-assigned or system-assigned identities, you can't transfer the resource to another Azure subscription or region. You must delete the managed identities before you move the resource. After the move, you must re-create the managed identities and assign them to the resource. For more information, see [Move resources to a new resource group or subscription](#).

- If you move a subscription from one directory to another, managed identities aren't preserved. You must move the resource and then [manually re-create the managed identities](#).
- Similar to user RBAC role assignments, [follow the principle of least privilege](#) when you grant a managed identity access to a resource.

External users

You can evaluate scenarios that involve setting up external users, customers, or partners so they can access resources. Determine whether these scenarios involve [Microsoft Entra B2B](#) or [Azure Active Directory B2C \(Azure AD B2C\)](#) configurations. For more information, see [Overview of Microsoft Entra External ID](#).

Design recommendations

Consider the following recommendations when designing the identity and access management of your applications.

OpenID Connect

If your application team uses continuous integration and continuous delivery (CI/CD) pipelines to deploy applications programmatically, configure OpenID Connect authentication to your Azure services. OpenID Connect uses a temporary, credential-free token to authenticate to Azure services. For more information, see [Workload identity federation](#).

If OpenID Connect isn't supported, create a service principal and assign the necessary permissions to allow infrastructure or application code to be deployed. For more information, see the training module, [Authenticate your Azure deployment pipeline by using service principals](#).

Attribute-based access control

To further restrict access and prevent unauthorized access to data, [use attribute-based access control \(ABAC\)](#) where supported, for example with Azure Blob Storage.

Virtual machine access

Where possible, use Microsoft Entra ID identities to control access to Azure virtual machines. Use Microsoft Entra ID instead of local authentication to provide access to virtual machines, taking advantage of Microsoft Entra Conditional Access, audit logging, and Microsoft Entra multifactor authentication (MFA). This configuration reduces the risk of attackers exploiting insecure local authentication services. For more information, see [Log into a Linux virtual machine in Azure by using Microsoft Entra ID and OpenSSH](#) and [Log into a Windows virtual machine in Azure using Microsoft Entra ID including passwordless](#).

Microsoft identity platform

- When developers build a cloud-native application, they should use the [Microsoft identity platform for developers](#) as the identity provider for their applications. The Microsoft identity platform provides an OpenID Connect standard-compliant authentication service that developers can use to authenticate several identity types including:
 - Work or school accounts, provisioned through Microsoft Entra ID

- Personal Microsoft accounts (Skype, Xbox, Outlook.com)
- Social or local accounts, by using Microsoft Entra ID
- The [Microsoft identity platform best practices and recommendations](#) checklist provides guidance on effectively integrating the application with the Microsoft identity platform.

Managed identities

- To enable access between Azure resources that don't need to use credentials, use managed identities.
- You shouldn't share credentials or managed identities among various environments or applications. For example, don't use identities for production resources and also in dev/test resources, even for the same application. Create separate credentials for each instance of an application to reduce the likelihood of a compromised test instance affecting production data. Separate credentials also make it easier to revoke credentials when they're no longer required.
- When there's a requirement to use managed identities at scale, use a user-assigned managed identity for each resource type in each region. This approach prevents a churn of identities. For example, Azure Monitor Agent requires a managed identity on monitored Azure VMs, which can cause Microsoft Entra ID to create (and delete) a substantial number of identities. You can create user-assigned managed identities once and share them across multiple VMs. Use [Azure Policy](#) to implement this recommendation.

Key Vault

- You can use Key Vault to manage the secrets, keys, certificates that applications use.
 - To [manage access](#) to secrets (data plane) and for administrative access (control plane), use RBAC.
 - To [control application access to Key Vault](#), use managed identities.
- You should use separate key vaults for each application environment (development, preproduction, production) in each region. Use RBAC to manage access to secrets, keys, and certificates (data plane operations) and access to Key Vault (control plane). Deploy key vaults that have application secrets into the application landing zones.

Microsoft Entra application proxy

- To access applications that use on-premises authentication remotely via Microsoft Entra ID, use [Microsoft Entra application proxy](#). Microsoft Entra application proxy provides secure remote access to on-premises web applications, including applications that use older authentication protocols. After a single sign-on to Microsoft Entra ID, users can access both cloud and on-premises applications via an external URL or an internal application portal.
 - You can deploy Microsoft Entra application proxy as a single instance into a Microsoft Entra ID tenant. Configuration requires at least the Application Administrator privileged Microsoft Entra ID role. If your organization uses subscription democratization as a role assignment model, application owners might not have the necessary permissions to configure Microsoft Entra application proxy. In this case, the platform team should configure Microsoft Entra application proxy for the application owner.
 - If you use CI/CD deployment pipelines with sufficient permissions, application owners can [configure Microsoft Entra application proxy by using the Microsoft Graph API](#).
- If the application uses legacy protocols, such as Kerberos, ensure that the application landing zone has network connectivity to domain controllers in the Microsoft identity platform subscription.

Next steps

[Resource organization](#)

Feedback

Was this page helpful?

 Yes

 No

Resource organization

Article • 05/29/2024

Use the resource organization design area to establish consistent patterns when you organize resources that you deploy to the cloud.

Design area review

Involved roles or functions: This design area requires support from one or more [cloud platform](#) and [cloud center of excellence](#) functions to make and implement decisions.

Scope: Resource organization decisions provide a foundation for all compliance-related design areas. When you plan your resource organization, you can establish consistent patterns for the following areas:

- Naming
- Tagging
- Subscription design
- Management group design

The initial scope of this exercise assumes a subscription design that aligns with the Azure landing zone conceptual architecture. Workload-level or application-level subscription and landing zone assignment supports separation of duties and subscription democratization requirements.

The following assumptions are the basis for workload subscription design pattern guidance:

- Your enterprise commits to long-term cloud operations.
- You need cloud management, security, and governance tooling to manage Azure, hybrid, or multicloud solutions.
- You have management or platform deployments in subscriptions and management groups that are separate from workload or application resources.

Multiple regions: The performance, reliability, and compliance of your cloud-based applications rely on Azure regions. Use the Azure global infrastructure to scale your applications when you need to. Regions provide the capacity to handle varying workloads. Whether you launch a new product or expand your user base, you must have the right resources in the right region to ensure agility, scalability, and high resiliency.

Use multiple regions for critical applications and services that require geo-disaster recovery capabilities. Multiple regions provide maximum resiliency. For information about how to select and operate in multiple regions, see [Select Azure regions](#).

Also consider the following factors when you deploy your workload in multiple regions:

- You can initially deploy in a single region and then expand to [multiple regions](#) in the future.
- To ensure consistency and manageability, properly organize resources when you adopt a multiregion design.
- Depending on your requirements and desired governance model, you can organize multiregion resources at various levels, such as the [management group](#), [subscription and resource group](#), [naming convention](#), and [tagging](#) levels.

New cloud environment: Start your cloud journey with a small set of subscriptions. For more information, see [Create your initial Azure subscriptions](#).

Existing cloud environment: If you have an existing cloud environment, consider the following guidance:

- If your current environment doesn't use [management groups](#), consider incorporating them. You can use management groups to manage policies, access, and compliance across subscriptions at scale.
- If your current environment uses management groups, see [Management groups](#). Use this guidance to help evaluate your implementation.
- If you have existing subscriptions in your current environment, ensure that you use them effectively. Subscriptions act as policy and management boundaries and scale units. For more information, see [Subscriptions](#).
- If you have existing resources in your current environment, see [Naming and tagging](#). Use this guidance to influence your tagging strategy and your naming conventions going forward.
- Use [Azure Policy](#) to establish and enforce consistency with taxonomic tags.

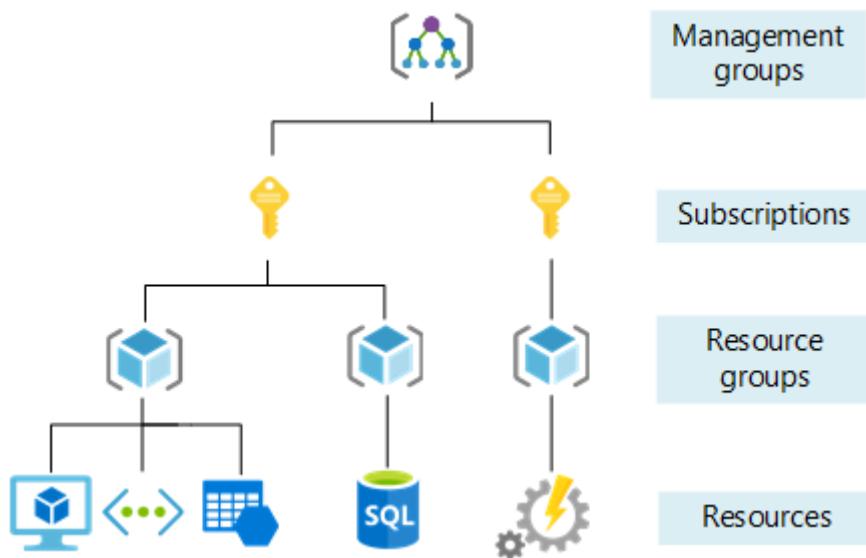
Design area overview

Cloud adoption journeys have various starting points and scale requirements. Some enterprises start with a few applications in the cloud and grow over time. Other enterprises must scale quickly to address business demands like a datacenter migration.

In both scenarios, when you plan your resource organization, you must factor in environment growth to accommodate more applications and services.

Simplify resource management across your environment to prepare for increased workload numbers and scale. Consider the foundational management groups and the subscription structure in your Azure landing zone design and implementation. Plan in advance to prevent future scaling constraints.

The resource organization design area explores techniques and technologies that help maintain proper resource topologies in cloud environments. The following diagram shows the four scope levels to organize Azure resources: management groups, subscriptions, resource groups, and resources.



Next steps

- [Management groups](#)
- [Subscriptions](#)

Feedback

Was this page helpful?

Yes

No

Management groups

Article • 05/29/2024

Use [management groups](#) to organize and govern your Azure subscriptions. As the number of your subscriptions increases, management groups provide critical structure to your Azure environment and make it easier to manage your subscriptions. Use the following guidance to establish an effective management group hierarchy and organize your subscriptions according to best practices.

Management group design considerations

Management group structures within a Microsoft Entra tenant support organizational mapping. Consider your management group structure thoroughly when your organization plans its Azure adoption at scale.

- Determine how your organization separates out services that specific teams own or operate.
- Determine whether you have specific functions that you need to keep separate for reasons like business requirements, operational requirements, regulatory requirements, data residency, data security, or data sovereignty compliance.
- Use management groups to aggregate policy and initiative assignments via Azure Policy.
- Enable Azure role-based access control (RBAC) authorization for management group operations to override the default authorization. By default, any principal, like a user principal or service principal, within a Microsoft Entra tenant can create new management groups. For more information, see [How to protect your resource hierarchy](#).

Also consider the following factors:

- A management group tree can support up to [six levels of depth](#). This limit doesn't include the tenant root level or the subscription level.
- All new subscriptions are placed under the tenant root management group by default.

For more information, see [Management groups](#).

Management group recommendations

- Keep the management group hierarchy reasonably flat, ideally with no more than three to four levels. This restriction reduces management overhead and complexity.
- Don't duplicate your organizational structure into a deeply nested management group hierarchy. Use management groups for policy assignment versus billing purposes. For this approach, use management groups for their intended purpose in the Azure landing zone conceptual architecture. This architecture provides Azure policies for workloads that require the same type of security and compliance under the same management group level.
- Create management groups under your root-level management group to represent the types of workloads that you host. These groups are based on the security, compliance, connectivity, and feature needs of the workloads. With this grouping structure, you can have a set of Azure policies applied at the management group level. Use this grouping structure for all workloads that require the same security, compliance, connectivity, and feature settings.
- Use resource tags to query and horizontally navigate across the management group hierarchy. You can use Azure Policy to enforce or append resource tags. Then you can group resources for search needs without having to use a complex management group hierarchy.
- Create a top-level sandbox management group so that you can immediately experiment with resources before you move them to production environments. The sandbox provides isolation from your development, test, and production environments.
- Create a platform management group under the root management group to support common platform policies and Azure role assignments. This grouping structure ensures that you can apply various policies to the subscriptions in your Azure foundation. This approach also centralizes the billing for common resources in one set of foundational subscriptions.
- Limit the number of Azure Policy assignments at the root management group scope. This limitation minimizes debugging inherited policies in lower-level management groups.
- Use policies to enforce compliance requirements either at the management group or subscription scope to achieve policy-driven governance.

- Ensure that only privileged users can operate management groups in the tenant. Enable Azure RBAC authorization in the management group [hierarchy settings](#) to refine user privileges. By default, all users can create their own management groups under the root management group.
- Configure a default, dedicated management group for new subscriptions. This group ensures that no subscriptions go under the root management group. This group is especially important if users have Microsoft Developer Network (MSDN) or Visual Studio benefits and subscriptions. A good candidate for this type of management group is a sandbox management group. For more information, see [Set a default management group](#).
- Don't create management groups for production, testing, and development environments. If necessary, separate these groups into different subscriptions in the same management group. For more information, see:
 - [Manage application development environments in Azure landing zones](#)
 - [Testing approach for enterprise-scale management groups](#)
- We recommended that you use the standard Azure landing zone management group structure for multiregion deployments. Don't create management groups solely to model different Azure regions. Don't alter or expand your management group structure based on region or multiregion usage.

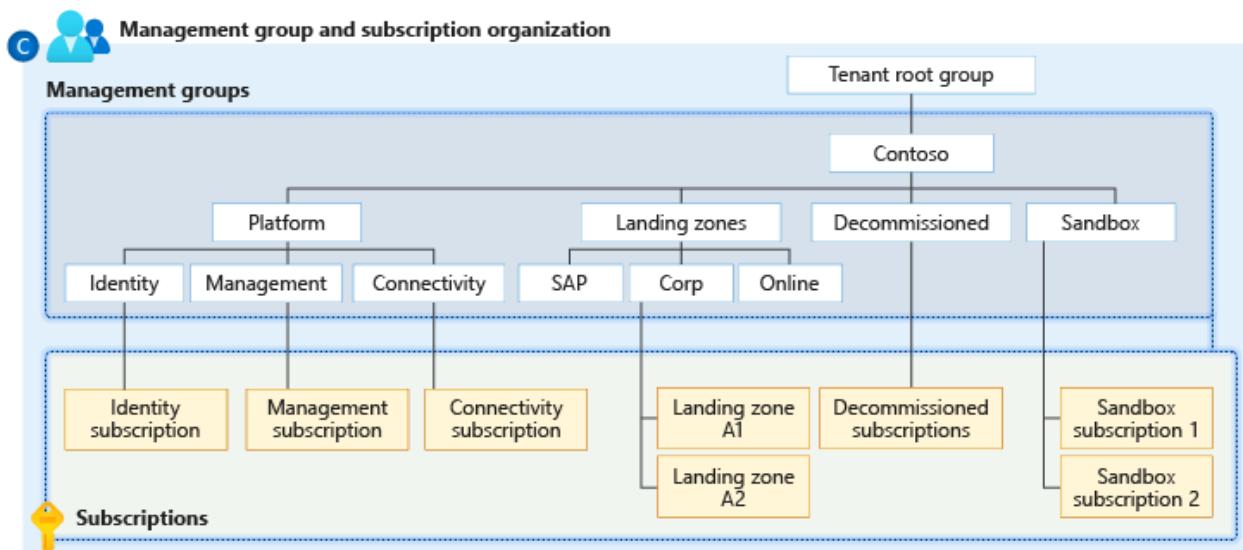
If you have location-based regulatory requirements, such as data residency, data security, or data sovereignty, then you should create a management group structure based on location. You can implement this structure at various levels. For more information, see [Modify an Azure landing zone architecture](#).

Management groups in the Azure landing zone accelerator and ALZ-Bicep repository

The following example shows a management group structure. The management groups in this example are in the Azure landing zone accelerator and the [management groups module of the ALZ-Bicep repo](#).

Note

You can modify the management group hierarchy in the Azure landing zone bicep module by editing [managementGroups.bicep](#).



[Expand table](#)

Management group	Description
Intermediate root management group	This management group is directly under the tenant root group. The organization provides this management group with a prefix so that they don't have to use the root group. The organization can move existing Azure subscriptions into the hierarchy. This approach also sets up future scenarios. This management group is a parent to all the other management groups created by the Azure landing zone accelerator.
Platform	This management group contains all the platform child management groups, like management, connectivity, and identity.
Management	This management group contains a dedicated subscription for management, monitoring, and security. This subscription hosts an Azure Monitor Logs workspace, including associated solutions and an Azure Automation account.
Connectivity	This management group contains a dedicated subscription for connectivity. This subscription hosts the Azure networking resources, like Azure Virtual WAN, Azure Firewall, and Azure DNS private zones, that the platform requires. You can use various resource groups to contain resources, such as virtual networks, firewall instances, and virtual network gateways, that are deployed in different regions. Some large deployments might have subscription quota restrictions for connectivity resources. You can create dedicated subscriptions in each region for their connectivity resources.
Identity	This management group contains a dedicated subscription for identity. This subscription is a placeholder for Active Directory Domain Services (AD DS) virtual machines (VMs) or Microsoft Entra Domain Services. You can use various resource groups to contain resources, such as virtual networks and VMs, that are deployed in different regions.

Management group	Description
	The subscription also enables AuthN or AuthZ for workloads within the landing zones. Assign specific Azure policies to harden and manage the resources in the identity subscription. Some large deployments might have subscription quota restrictions for connectivity resources. You can create dedicated subscriptions in each region for their connectivity resources.
Landing zones	The parent management group that contains all the landing zone child management groups. It has workload-agnostic Azure policies assigned to ensure that workloads are secure and compliant.
Online	The dedicated management group for online landing zones. This group is for workloads that might require direct internet inbound or outbound connectivity or for workloads that might not require a virtual network.
Corp	The dedicated management group for corporate landing zones. This group is for workloads that require connectivity or hybrid connectivity with the corporate network via the hub in the connectivity subscription.
Sandboxes	The dedicated management group for subscriptions. An organization uses sandboxes for testing and exploration. These subscriptions are securely isolated from the corporate and online landing zones. Sandboxes also have a less restrictive set of policies assigned to enable testing, exploration, and configuration of Azure services.
Decommissioned	The dedicated management group for canceled landing zones. You move canceled landing zones to this management group, and then Azure deletes them after 30-60 days.

 **Note**

For many organizations, the default `Corp` and `Online` management groups provide an ideal starting point. Some organizations need to add more management groups.

If you want to change the management group hierarchy, see [Tailor the Azure landing zone architecture to meet requirements](#).

Permissions for the Azure landing zone accelerator

The Azure landing zone accelerator:

- Requires a dedicated service principal name (SPN) to run management group operations, subscription management operations, and role assignments. Use an SPN to reduce the number of users that have elevated rights and follow least-privilege guidelines.
- Requires the User Access Administrator role at the root management group scope to grant the SPN access at the root level. After the SPN has permissions, you can safely remove the User Access Administrator role. This approach ensures that only the SPN is connected to the User Access Administrator role.
- Requires the Contributor role for the SPN previously mentioned at the root management group scope, which allows tenant-level operations. This permission level ensures that you can use the SPN to deploy and manage resources to any subscription within your organization.

Next step

Learn how to use subscriptions when you plan a large-scale Azure adoption.

[Subscriptions](#)

Feedback

Was this page helpful?

 Yes

 No

Subscription considerations and recommendations

Article • 11/28/2024

Subscriptions are a unit of management, billing, and scale within Azure. They play a critical role when you design for large-scale Azure adoption. This article helps you capture subscription requirements and design target subscriptions based on critical factors that vary depending on:

- Environment types
- Ownership and governance models
- Organizational structures
- Application portfolios
- Regions

💡 Tip

For more information about subscriptions, see the YouTube video: [Azure landing zones - How many subscriptions should I use in Azure? ↗](#)

ⓘ Note

If you use Enterprise Agreements, Microsoft Customer Agreements (Enterprise), or Microsoft Partner Agreements (CSP), review the subscription limits in [Billing accounts and scopes in the Azure portal](#).

Subscription considerations

The following sections contain considerations to help you plan and create subscriptions for Azure.

Organization and governance design considerations

- Subscriptions serve as boundaries for Azure Policy assignments.

For example, secure workloads like Payment Card Industry (PCI) workloads typically require other policies in order to achieve compliance. Instead of using a management group to collate workloads that require PCI compliance, you can

achieve the same isolation with a subscription, without having too many management groups with a few subscriptions.

If you need to group together many subscriptions of the same workload archetype, create them under a management group.

- Subscriptions serve as a scale unit so component workloads can scale within platform [subscription limits](#). Make sure you consider subscription resource limits as you design your workloads.
- Subscriptions provide a management boundary for governance and isolation that clearly separates concerns.
- Create separate platform subscriptions for management (monitoring), connectivity, and identity when they're required.
 - Establish a dedicated management subscription in your platform management group to support global management capabilities like Azure Monitor Logs workspaces and Azure Automation runbooks.
 - Establish a dedicated identity subscription in your platform management group to host Windows Server Active Directory domain controllers when needed.
 - Establish a dedicated connectivity subscription in your platform management group to host an Azure Virtual WAN hub, private Domain Name System (DNS), Azure ExpressRoute circuit, and other networking resources. A dedicated subscription ensures that all your foundation network resources are billed together and isolated from other workloads.
 - Use subscriptions as a democratized unit of management that aligns with your business needs and priorities.
- Use manual processes to limit Microsoft Entra tenants to only Enterprise Agreement enrollment subscriptions. When you use a manual process, you can't create Microsoft Developer Network (MSDN) subscriptions at the root management group scope.

For support, submit an [Azure support ticket](#).

For information about subscription transfers between Azure billing offers, see [Azure subscription and reservation transfer hub](#).

Multiple region considerations

Important

Subscriptions aren't tied to a specific region, and you can treat them as global subscriptions. They're logical constructs to provide billing, governance, security, and identity controls for Azure resources that are contained within them. Therefore, you don't need a separate subscription for each region.

- You can adopt a multiregion approach at the single workload level for scaling or geo-disaster recovery or at a global level (different workloads in different regions).
- A single subscription can contain resources from different regions, depending on the requirements and architecture.
- In a geo-disaster recovery context, you can use the same subscription to contain resources from primary and secondary regions because they're logically part of the same workload.
- You can deploy different environments for the same workload in different regions to optimize costs and resource availability.
- In a subscription that contains resources from multiple regions, you can use resource groups to organize and contain resources by region.

Quota and capacity design considerations

Azure regions might have a finite number of resources. As a result, you should track the available capacity and SKUs for Azure adoptions with several resources.

- Consider [limits and quotas](#) within the Azure platform for each service that your workloads require.
- Consider the availability of required SKUs within your chosen Azure regions. For example, new features might be available only in certain regions. The availability of certain SKUs for given resources like virtual machines (VMs) can vary from one region to another.
- Consider that subscription quotas aren't capacity guarantees and are applied on a per-region basis.

For virtual machine capacity reservations, see [On-demand capacity reservation](#).

- Consider reusing unused or decommissioned subscriptions. For more information, see [Create or reuse Azure subscriptions](#).

Tenant transfer restriction design considerations

Each Azure subscription is linked to a single Microsoft Entra tenant, which acts as an identity provider (IdP) for your Azure subscription. Use the Microsoft Entra tenant to authenticate users, services, and devices.

When any user has the required permissions, they can change the Microsoft Entra tenant that's linked to your Azure subscription. For more information, see:

- [Associate or add an Azure subscription to your Microsoft Entra tenant](#)
- [Transfer an Azure subscription to a different Microsoft Entra directory](#)

ⓘ Note

You can't transfer to a different Microsoft Entra tenant for Azure Cloud Solution Provider (CSP) subscriptions.

For Azure landing zones, you can set requirements to prevent users from transferring subscriptions to your organization's Microsoft Entra tenant. For more information, see [Manage Azure subscription policies](#).

Configure your subscription policy by providing a list of [exempted users](#). Exempted users are permitted to bypass restrictions that are set in the policy.

ⓘ Important

An exempted users list isn't an [Azure Policy](#).

- Consider whether you should allow users that have [Visual Studio or MSDN Azure subscriptions](#) to transfer their subscription to or from your Microsoft Entra tenant.
- Only users with the Microsoft Entra [Global Administrator](#) role can configure tenant transfer settings. These users must have [elevated access](#) to change the policy.
 - You can only specify individual user accounts as [exempted users](#), not Microsoft Entra groups.

ⓘ Important

Microsoft recommends that you use roles with the fewest permissions. This helps improve security for your organization. Global Administrator is a highly privileged

role that should be limited to emergency scenarios when you can't use an existing role.

- All users with access to Azure can view the policy that's defined for your Microsoft Entra tenant.
 - Users can't view your [exempted users](#) list.
 - Users can view the global administrators within your Microsoft Entra tenant.
- Azure subscriptions that you transfer into a Microsoft Entra tenant are placed into the [default management group](#) for that tenant.
- If your organization approves, your application team can define a process to allow Azure subscriptions to be transferred to or from a Microsoft Entra tenant.

Cost management design considerations

Every large enterprise organization has the challenge of managing cost transparency. This section explores key aspects to achieve cost transparency across large Azure environments.

- You might need to share chargeback models, like App Service Environment and Azure Kubernetes Service (AKS), to achieve higher density. Chargeback models can affect shared platform as a service (PaaS) resources.
- Use a shutdown schedule for nonproduction workloads to optimize costs.
- Use [Azure Advisor](#) to get recommendations for optimizing costs.
- Establish a chargeback model for better distribution of cost across your organization.
- Implement policy so that users can't deploy unauthorized resources in your organization's environment.
- Establish a regular schedule and cadence to review cost and rightsize resources for workloads.

Subscription recommendations

The following sections contain recommendations to help you plan and create subscriptions for Azure.

Organization and governance recommendations

- Treat subscriptions as a unit of management that aligns with your business needs and priorities.
- Inform subscription owners of their roles and responsibilities.
 - Do a quarterly or yearly access review for Microsoft Entra Privileged Identity Management (PIM) to ensure that privileges don't proliferate when users move within your organization.
 - Take full ownership of budget spending and resources.
 - Ensure policy compliance and remediate when necessary.
- When you identify requirements for new subscriptions, reference the following principles:
 - **Scale limits:** Subscriptions serve as a scale unit for component workloads to scale within platform subscription limits. Large specialized workloads, like high-performance computing, IoT, and SAP, should use separate subscriptions to avoid running up against these limits.
 - **Management boundary:** Subscriptions provide a management boundary for governance and isolation, which allows a clear separation of concerns. Various environments, such as development, test, and production environments, are often removed from a management perspective.
 - **Policy boundary:** Subscriptions serve as a boundary for the Azure Policy assignments. For example, secure workloads like PCI workloads typically require other policies in order to achieve compliance. The overhead doesn't get considered if you use a separate subscription. Development environments have more relaxed policy requirements than production environments.
 - **Target network topology:** You can't share virtual networks across subscriptions, but you can connect them with different technologies like virtual network peering or ExpressRoute. When you decide if you need a new subscription, consider which workloads need to communicate with each other.
- Group subscriptions together under management groups, which are aligned with your management group structure and policy requirements. Group subscriptions to ensure that subscriptions with the same set of policies and Azure role assignments come from the same management group.

- Establish a dedicated management subscription in your `Platform` management group to support global management capabilities like Azure Monitor Logs workspaces and Automation runbooks.
- Establish a dedicated identity subscription in your `Platform` management group to host Windows Server Active Directory domain controllers when necessary.
- Establish a dedicated connectivity subscription in your `Platform` management group to host a Virtual WAN hub, private DNS, ExpressRoute circuit, and other networking resources. A dedicated subscription ensures that all your foundation network resources are billed together and isolated from other workloads.
- Avoid a rigid subscription model. Instead, use a set of flexible criteria to group subscriptions across your organization. This flexibility ensures that as your organization's structure and workload composition changes, you can create new subscription groups instead of using a fixed set of existing subscriptions. One size doesn't fit all for subscriptions, and what works for one business unit might not work for another. Some applications might coexist within the same landing zone subscription, while others might require their own subscription.

For more information, see [Handle dev/test/production workload landing zones](#).

Multiple regions recommendations

- Create additional subscriptions for each region only if you have region-specific governance and management requirements, for example data sovereignty or to scale beyond quota limits.
- If scaling isn't a concern for a geo-disaster recovery environment that spans multiple regions, use the same subscription for the primary and secondary region resources. Some Azure services, depending on the business continuity and disaster recovery (BCDR) strategy and tools that you adopt, might need to use the same subscription. In an active-active scenario, where deployments are independently managed or have different life cycles, we recommend that you use different subscriptions.
- The region where you create a resource group and the region of the contained resources should match so they don't affect resilience and reliability.
- A single resource group shouldn't contain resources from different regions. This approach can lead to problems with resource management and availability.

Quota and capacity recommendations

- Use subscriptions as scale units, and scale out resources and subscriptions as required. Your workload can then use the required resources for scaling out without reaching subscription limits in the Azure platform.
- Use capacity reservations to manage capacity in some regions. Your workload can then have the required capacity for high demand resources in a specific region.
- Establish a dashboard that has custom views to monitor used capacity levels, and set up alerts if capacity approaches critical levels, such as 90% CPU usage.
- Raise support requests for quota increases under subscription provisioning, such as for total available VM cores within a subscription. Ensure that your quota limits are set before your workloads exceed the default limits.
- Ensure that any required services and features are available within your chosen deployment regions.

Automation recommendations

- Build a subscription vending process to automate the creation of subscriptions for application teams via a request workflow. For more information, see [Subscription vending](#).

Tenant transfer restriction recommendations

- Configure the following settings to prevent users from transferring Azure subscriptions to or from your Microsoft Entra tenant:
 - Set *Subscription leaving Microsoft Entra directory* to `Permit no one`.
 - Set *Subscription entering Microsoft Entra directory* to `Permit no one`.
- Configure a limited list of [exempted users](#).
 - Include members from an Azure platform operations team.
 - Include break-glass accounts in the list of [exempted users](#).

Next step

[Adopt policy-driven guardrails](#)

Feedback

Was this page helpful?

 Yes

 No

Manage application development environments in Azure landing zones

Article • 12/12/2023

This article describes how cloud platform teams can implement guardrails to manage application environments in Azure landing zones. It also explains how to align various application development environments with their framework. A key aspect in creating the proper environment is placing subscriptions in the appropriate management groups.

Set the foundation

Development teams require the ability to iterate quickly, and cloud governance and platform teams need to manage organizational risk, compliance, and security at scale. You can properly manage application environments by focusing on two key [Azure landing zone design principles](#): policy-driven governance and subscription democratization. These principles provide foundational guardrails and describe how to delegate controls to application teams. The application teams use [Azure Well-Architected Framework guidance](#) to design their workload. They deploy and manage their own landing zone resources, and the platform team controls the resources by assigning Azure policies.

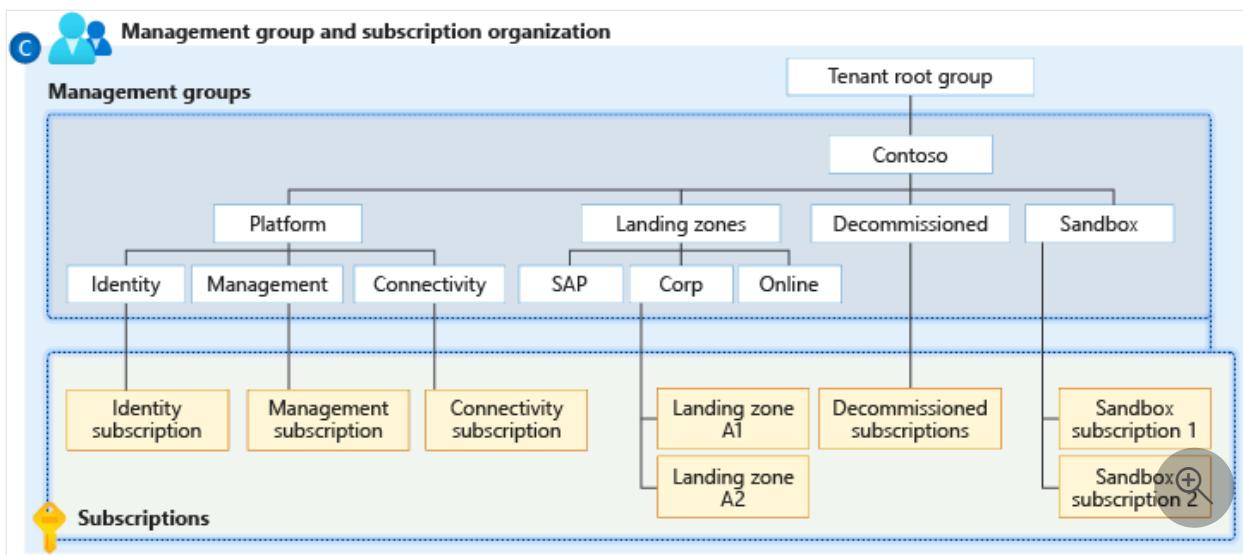
It's important to provide sandbox resources for *semi-governed* resources, so application teams can experiment with technologies and capabilities.

When application owners use [subscription vending](#) or other subscription creation processes, they must know how to request subscriptions for multiple development environments.

This article describes the Azure landing zone, including the management groups, policies, and shared platform architecture, and the workload or application landing zone.

Note

The guidance in this article is only for workload or application landing zones. For testing and environment segregation for the Azure landing zone platform itself, see [Testing approach for Azure landing zones](#), which describes the canary approach.



In practice, you can use any number and type of phased environment. This article references the following phased environments.

[Expand table](#)

Environment	Description	Management group
Sandbox	The environment that's used for rapid innovation of prototypes but not production-bound configurations	Sandbox management group
Development	The environment that's used to build potential release candidates	Archetype management group, like <i>corp</i> or <i>online</i>
Test	The environment that's used to perform testing, including unit testing, user acceptance testing, and quality assurance testing	Archetype management group, like <i>corp</i> or <i>online</i>
Production	The environment that's used to deliver value to customers	Archetype management group, like <i>corp</i> or <i>online</i>

For more information, see the videos [Handling development, testing, and production environments for application workloads](#) and [How many subscriptions should I use in Azure?](#)

Environments, subscriptions, and management groups

As a prerequisite to this section, see [Resource organization design area](#).

You must properly organize your subscriptions when you adopt Azure landing zone practices. Ideally, each application environment should have its own subscription. This

method provides security and policy controls that keep the environments isolated. It contains potential problems to one environment.

Separate subscriptions have the same policies on the archetype level. If needed, application owners can assign subscription-specific policies to enforce application and environment-specific behavior.

Some application architectures require that services are shared among environments. If that's the case, you can use a single subscription for multiple environments. We recommend that workload owners work with cloud platform teams to determine if a single subscription for multiple environments is needed.

Use a single subscription for multiple application environments if:

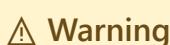
- The environments can't be isolated in their own subscriptions.
- The environments have the same teams assigned to functional roles, such as network operators.
- The environments can use the same policies.

If an application or service workload needs to be in a single subscription, and you need to make changes to the policies that apply to each environment, you can:

- Create a new *archetype-aligned* management group beneath the landing zones management group. For more information, see [Management group hierarchy](#) in this article.
- Use sandbox subscriptions for development activities. Sandboxes have a less restrictive policy set.
- Use policies that are applied at the subscription level instead of the management group level. You can add tags in the policy definitions to help filter and apply policies to the correct environment. You can also assign policies to or exclude them from specific resource groups.

You can assign policies during the subscription creation process as part of [subscription vending](#).

For policies that you implement to help control costs, apply the policy definition at a subscription level where required. Or you can make the landing zone owner responsible for costs, which provides true autonomy. For more information, see [Platform automation and DevOps](#).



Unlike policies and controls at the management group level, subscription-based policies and tags can be changed by individuals with elevated permissions to the subscription. Administrators with appropriate roles can bypass these controls by excluding policies, modifying policies, or changing the tags on resources.

As a result, you shouldn't apply tags in the definitions of security-focused policies. In addition, don't assign permissions as *always active* for the following actions:

- `Microsoft.Authorization/policyAssignments/*`
- `Microsoft.Authorization/policyDefinitions/*`
- `Microsoft.Authorization/policyExemptions/*`
- `Microsoft.Authorization/policySetDefinitions/*`

You can control these actions by using Privileged Identity Management (PIM).

Management group hierarchy

Avoid complicated management group hierarchies. They can require frequent amendment, scale inefficiently, and lack value. To avoid these potential problems, Azure landing zone management groups are workload archetype-aligned. For more information, see [Management group and subscription organization](#).

Archetype-aligned means that management groups are only created for specific workload archetypes. For example, in the conceptual architecture, the *landing zones* management group has *corp* and *online* child management groups. These child management groups align with distinct archetype patterns for the workloads that they hold. The child management groups focus on hybrid connectivity (VPN/Azure ExpressRoute) requirements, such as internal only versus public-facing applications and services.

Excluding sandbox environments, various application environments should use the same archetype for deployment. Even if environments are divided across several subscriptions, they're held within the same single management group (*corp* or *online*), based on the management group archetype and the requirements.

You can use [sandbox subscriptions](#) for unstructured development, such as personal labs or for a workload that doesn't have an archetype. An application or service workload team uses a sandbox management group to test various Azure services to determine what works best for their requirements. After they decide on services, they can provision a landing zone (in the correct workload archetype-aligned management group in the *landing zones* management group hierarchy) for the team.

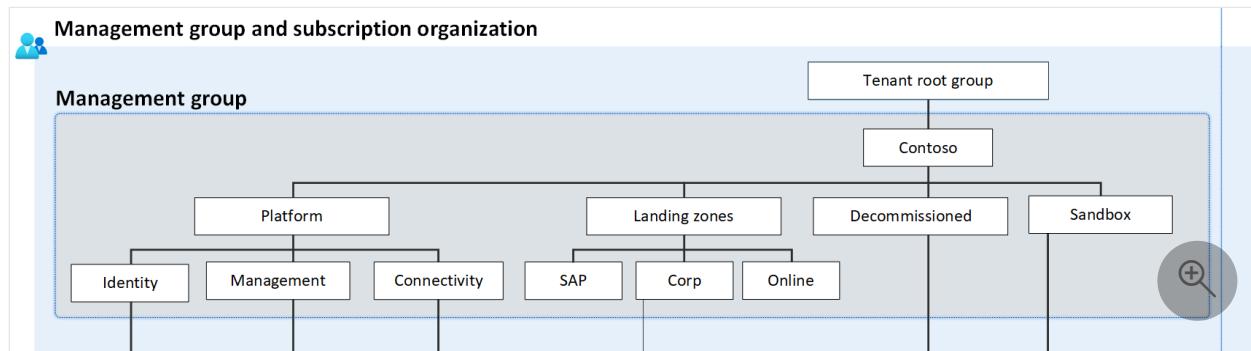
The sandbox environments can be used for specific applications, or a workload team can use them for experimentation.

For more information, see:

- [Management groups](#).
- [Resource organization design area](#).
- [Tailor the Azure landing zone architecture to meet requirements](#).

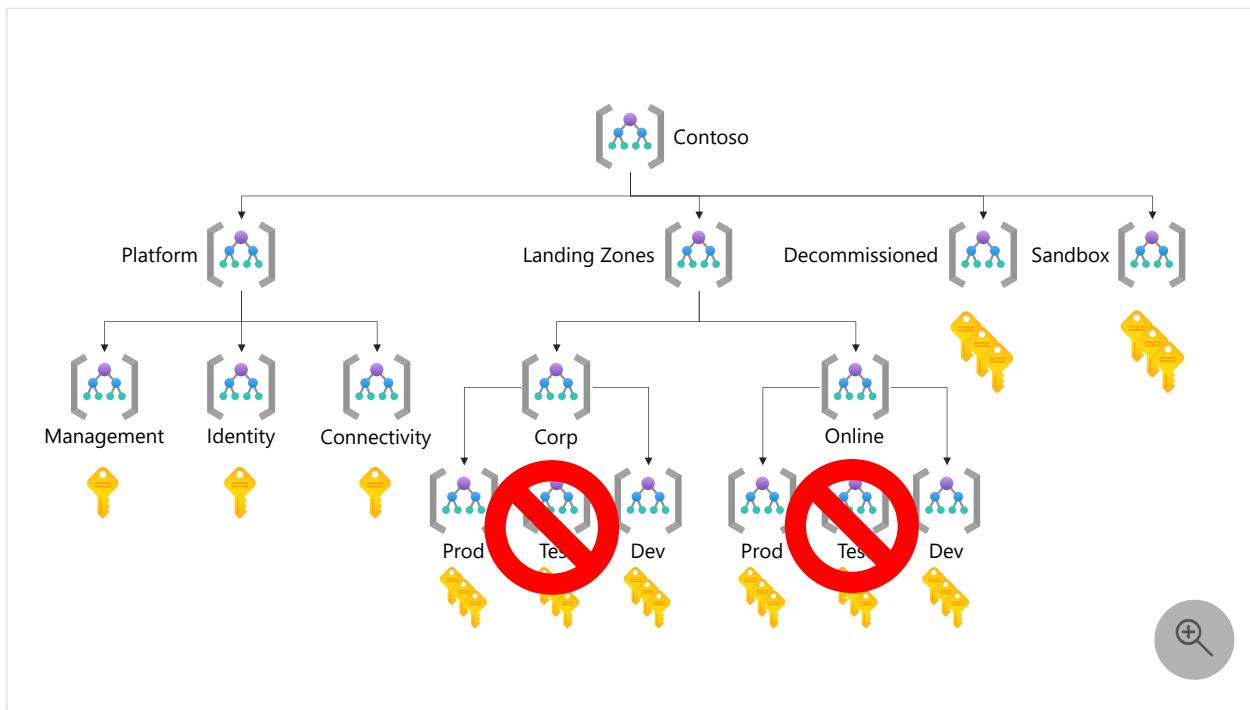
Environment-based management group challenges

Management groups for environments within archetypes can add management overhead and provide minimal value.



The *landing zones* management group should have universal policies that enforce guardrails for both *corp* and *online* child management groups. *Corp* and *online* have unique policies that enforce company guidelines related to public and private-facing workloads.

Many organizations create separate management groups for workload software development lifecycle (SDLC) environments to assign environmental policies and controls. In practice, this method creates more challenges for workload teams than it solves. SDLC environments shouldn't have different policies, so we don't recommend separate management groups.



Application owners can change the topology or resource configuration of a workload to align to policies in multiple SDLC environments that it's promoted through. This method increases risk. Rules that are specific to each environment can result in a poor development experience for developer and quality assurance teams. Problems can also arise if an application has one set of guardrail policies that work in one environment, and the application is exposed to a different set of policies later in its promotion cycle. You might have to make adjustments to an application if controls change.

To prevent this extra work, create consistent policies throughout the promotion of code in SDLC environments. You shouldn't create policies for each environment, but instead provide a consistent set for all development environments, excluding sandbox environments.

For example, imagine an organization defines a policy that requires storage accounts to be configured with specific firewall rules to prevent ingress from public networks. Instead, the storage accounts use private endpoints inside of the Azure landing zone networks for communication. If the development environment doesn't have such a policy, testing the workload doesn't find a misconfiguration of the storage account that allows public access. The test deployments work in the development environment and are iterated on. When the solution is promoted to another environment that has the storage account policy, the deployment fails because of the enforced policy.

As a result, the application development team must rework their deployment and architecture, after already investing significant effort. This example demonstrates how different policies in various environments can create problems.

Note

The following equation demonstrates why a separate management group for each environment or workload doesn't scale well: $N \text{ workloads} \times Z \text{ management groups} = \text{total management groups}$.

If an organization has 30 workloads that each require a management group and a child management group for development, testing, and production environments, the organization is left with:

N = the number of workloads/apps = 30

Z = the number of management groups for the workload and environments (1 for the workload + 3 for the environments) = 4

$N (30) \times Z (4) = 120$ total management groups

Application owners might need policies to apply differently to multiple environments. For example, application owners might require backup configurations for production environments but not for other environments.

Some policies can be enabled as audit policies at the management group level. Application teams determine how to implement the control. This method doesn't prevent deployments, but it creates awareness and enables application teams to manage their unique needs. They can then create sublevel policies or incorporate these requirements into their infrastructure as code (IaC) deployment modules.

In this shared responsibility model, the platform team audits practices, and the application team manages the implementation. This model can improve the agility of deployments.

Platform operators must work with each application or service workload team (landing zone owners) to understand their requirements. The platform operators can provide subscriptions based on the application requirements and plans. The platform operators might also decide to designate *product lines* for various types of workloads so that they can build subscription creation processes and tooling based on common requirements from application or service workload teams.

Scenario: Virtual machine (VM) based workloads

Early workloads in Azure landing zones are often made up of Azure VMs. You might deploy these workloads in Azure or migrate them from existing datacenters.

Instead of deploying VMs to multiple environments in a single subscription, you can:

- Establish subscriptions for each application environment, and place them all in the same archetype management group.
- Deploy a virtual network for each application environment in the appropriate subscription. You can determine the virtual network size based on the size of the application environment.
- Deploy the VMs to their appropriate subscription. VMs can use different SKUs and different availability configurations for each environment, if appropriate.

Various application environment resources are protected by different access controls. As a result, when application developers set up deployment pipelines, each pipeline's identity can be limited to the environment. This configuration helps to protect the environments from accidental deployments.

Scenario: Azure App Service

A workload with environmental subscriptions that use [App Service](#) can create challenges. For application developers, an [App Service best practice](#) is to use [deployment slots](#) to help manage changes and updates to a web app.

However, this feature can only be used with the app that's on the App Service plan, which can only live within a single subscription. If the platform operators mandate that the application owners use separate subscriptions for development, testing, and production environments, the application deployment lifecycle might be more difficult to manage.

For this example, the best option is a single subscription for the application or service workload. Application owners can use Azure role-based access control (RBAC) with [PIM](#) at the resource group level for increased security.

Next steps

- [Tailor the Azure landing zone architecture to meet requirements](#)
- [Testing approach for Azure landing zones](#)
- [Landing zone sandbox environments](#)

Feedback

Was this page helpful?

 Yes

 No

Modify an Azure landing zone architecture to meet requirements across multiple locations

Article • 12/06/2023

Organizations in many industries are subject to regulatory requirements, including data residency, data security, and data sovereignty requirements. Some organizations need to comply with conflicting regulations across multiple geographic locations. In this case, they need to modify their Azure landing zone architecture in accordance with all the applicable regulations.

For example, there might be two conflicting regulations, regulation A and regulation B. Regulation A might require data residency in country or region A, and regulation B might require data residency in country or region B.

Such regulatory conflicts can apply to:

- Multinational organizations, such as multinational corporations or non-governmental organizations (NGOs), that must comply with local regulations in the countries or regions that they operate in.
- Independent software vendors (ISVs) that provide solutions to organizations in multiple locations, and the solution must comply with the local regulations in each location.
- ISVs that provide solutions to multinational organizations that need to comply with the local regulations of each country or region that they operate in.

If you only need to meet a single set of regulatory requirements, see [Tailor the Azure landing zone architecture to meet requirements](#).

Regulatory considerations

Regulatory requirements are typically related to data protection, data residency, data transfers, isolation, or personnel clearance. These requirements can conflict among multiple geographic locations. For example, a European Union (EU) regulation might require data residency in an EU country, while a United Kingdom regulation might require data residency in the United Kingdom.

If regulations lead to conflicting policy controls, you must adjust the Azure landing zone architecture and policy assignments accordingly. For more information, see [the section in this article, Scenarios that require modification](#).

When multiple regulations apply, you don't need to modify the Azure landing zone architecture if:

- Multiple regulations require identical Azure Policy assignments.
- The controls in one regulation are a superset of another regulation. The superset controls automatically apply to both regulations.
- The controls in multiple regulations don't overlap. When you implement multiple control sets, a single implementation covers all regulations. Azure Policy assignments are complementary.
- Various regulations have different types of implementation. From a regulatory perspective, it doesn't matter which implementation you choose. For example, there might be two regulations that each have a different authorization model, but both authorization models are acceptable. You can choose the implementation that best fits your organization.

💡 Tip

You should strive to have as few policy assignments and exceptions or exemptions as possible.

Considerations for ISVs

There are three [deployment models for ISVs](#).

- **Pure software as a service (SaaS)**: The ISV provides the solution as a service.
- **Customer deployed**: The customer deploys the solution in their own environment.
- **Dual-deployment SaaS**: This model combines the customer-deployed model and the pure SaaS model.

In a *pure SaaS model*, the ISV is responsible for managing compliance on behalf of the customer. The ISV must demonstrate compliance to the customer and potentially to auditors or regulators. If you use the SaaS model, your architecture might be subject to multiple regulations that can conflict. The ISV must manage compliance for these

various regulations. For more information, see [the section in this article, Scenarios that require modification](#).

In a *customer-deployed model*, the customer is responsible for managing compliance. For this model, the ISV doesn't need to modify the landing zones. However, the solution is deployed in a landing zone that the customer deploys, including any policy controls and custom policies.

💡 Tip

ISVs can target policy initiatives at particular compliance requirements to test a solution. This practice can help minimize the chance of conflicts with policies that customers use to meet their compliance requirements.

In a *dual-deployment SaaS model*, all the considerations for the customer-deployed and pure SaaS model apply.

Considerations for multinational organizations

Multinational organizations use various structures to organize their IT governance.

- **Decentralized structure:** IT functions are governed locally in each geographic location.
- **Centralized structure:** IT functions are governed from a centralized place, typically the organization's headquarters.
- **Hybrid structure:** Global IT functions are provided centrally, while IT functions required only locally are governed in each geographic location.

In a *decentralized* scenario, the local IT team is responsible for managing compliance and can tailor their landing zone accordingly.

In a *centralized* scenario, the central IT team is responsible for managing compliance and must ensure that solutions meet the local compliance requirements of all the geographic locations where the multinational organization operates. The compliance requirements of various geographic locations can conflict, and it might be necessary to modify landing zones.

In a *hybrid* scenario, the considerations for both the decentralized and centralized scenarios apply. The centralized organization provides solutions that the local organizations need to deploy in their environment. The centralized organization also tests that those solutions deploy in all landing zones of the local organizations.

Scenarios that require modification

You might need to modify landing zones if there are conflicting policy sets that are assigned to various deployments. There might be multiple solutions or a single solution that need to be made available to various geographic locations or data classifications.

The amount of modification that's required depends on the level of isolation that the regulation calls for. The more conditions that a regulation has, the more the landing zone needs to be modified. For example, if regulations require conditions like cleared personnel, various identity providers or directories, separate management infrastructure, or separate connectivity infrastructure, the landing zone requires extensive modification. If regulations only require that the application and connectivity infrastructure be isolated, the landing zone needs minimal modification.

Microsoft Entra tenants

We recommend [using a single Microsoft Entra tenant](#) for most scenarios, including multinational scenarios. However, there are scenarios where you might prefer or require multiple Microsoft Entra tenants, such as:

- If you need to [separate the corporate Microsoft Entra tenant from the SaaS Microsoft Entra tenant](#) to improve security and create clear boundaries between the product and business operations.
- If conflicting regulations apply, and you need separate Microsoft Entra tenants for different regulatory regimes. For example, regulations might have clearance and nationality requirements that need complete isolation between [Microsoft Entra tenants or data residency requirements that require separate tenants](#). Common scenarios include an ISV that needs to deploy isolated instances of a SaaS solution or a multinational organization that needs to deploy isolated instances of the same solution.

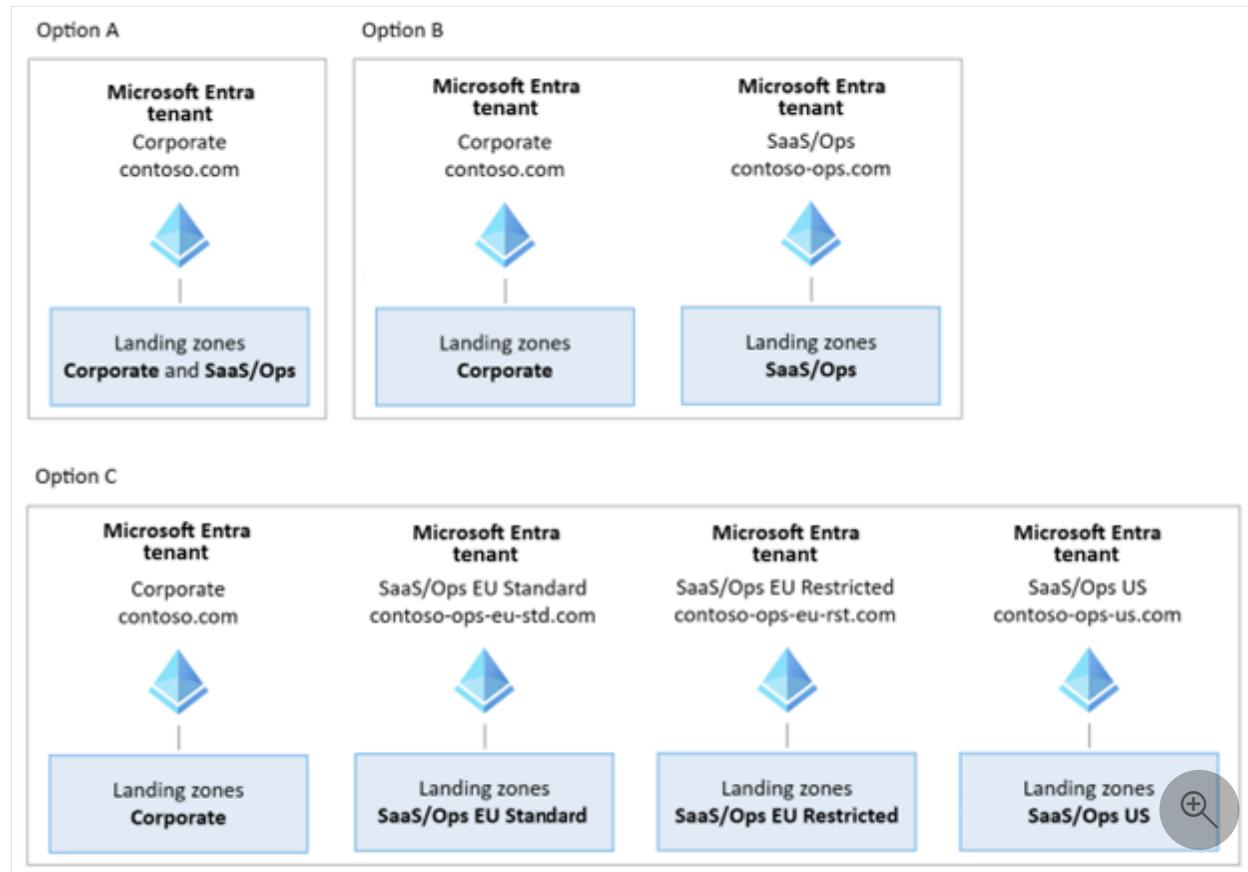
When you collaborate across multiple Microsoft Entra tenants, you need to carefully plan for significant challenges and needs. Create only the minimum number of Microsoft Entra tenants that you need to meet operational or regulatory requirements. You can use management groups and Azure role-based access control (RBAC) to govern the access to subscriptions and resources under a single tenant, as described in the next section.



Tip

The Microsoft Entra tenant that you select for your landing zone doesn't affect your application-level authentication. You can still use other identity providers regardless of which tenant you choose. For public sector customers and customers in regulated industries, end-user identities are typically provided when you integrate with an approved identity provider, such as a government-owned or certified identity provider.

The following diagrams show options that you can use to organize Microsoft Entra tenants.



Tip

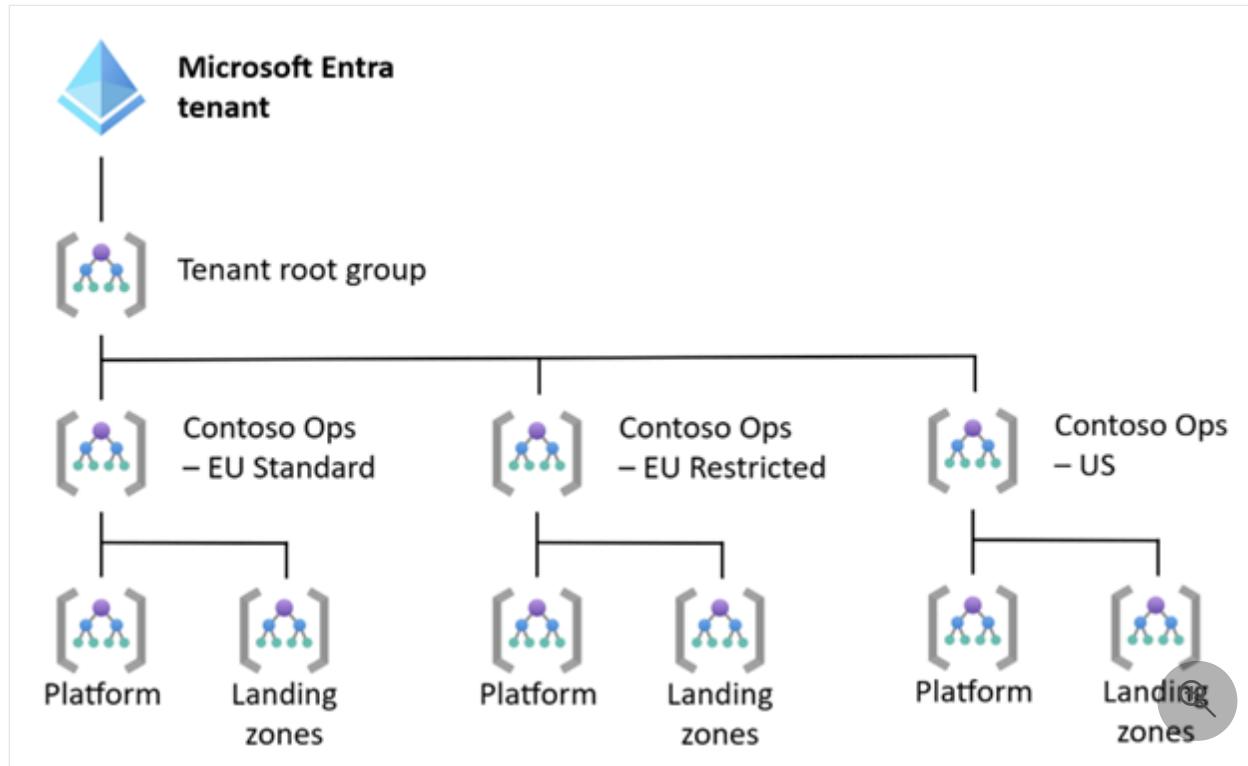
If you have multiple Microsoft Entra tenants to meet regulatory requirements, name the tenants based on the geographic location rather than specific regulations, for example `contoso-ops-us.com` in the example diagram.

For more information, see [Azure landing zones and multiple Microsoft Entra tenants](#) and [ISV considerations for Azure landing zones](#).

Management groups

If you don't need separate Microsoft Entra tenants in order to provide strict isolation, you should deploy multiple Azure landing zones in a single Microsoft Entra tenant. You can adjust the management group hierarchy to address the requirements of conflicting regulations.

You can deploy a full landing zone architecture for each set of regulations that you want to separate. This model requires the least amount of customization and enables you to take advantage of existing automation for deployment.



ⓘ Note

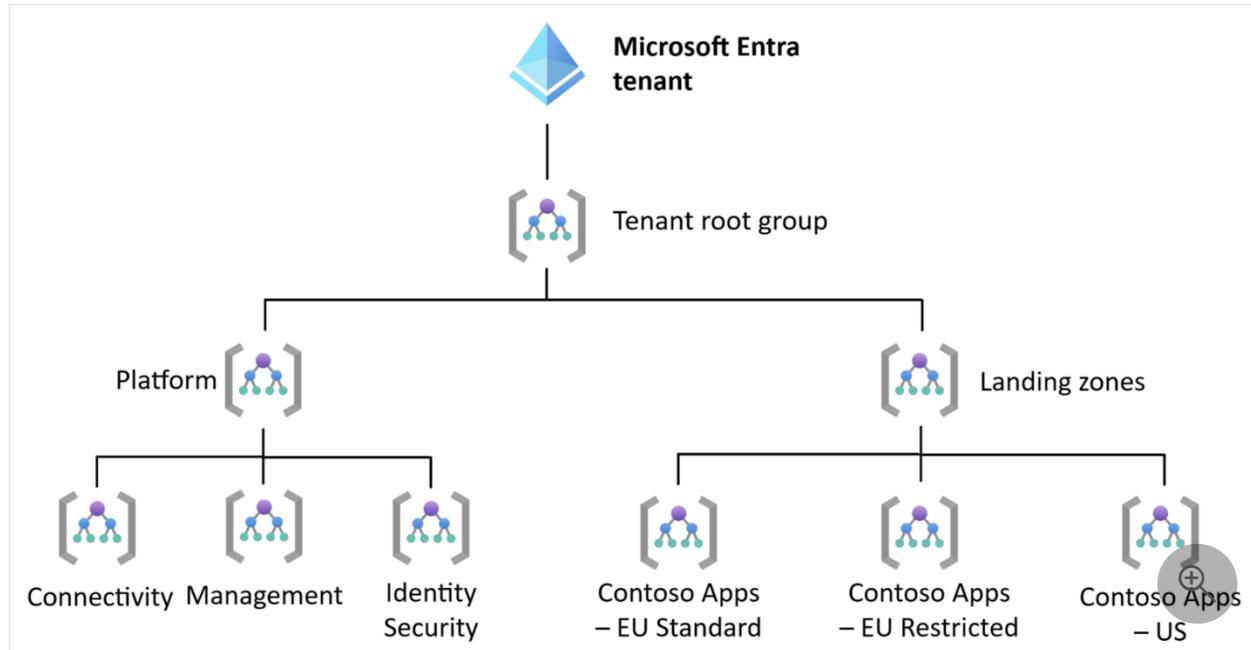
This diagram doesn't show all management groups.

Share the platform management group

If regulation allows, the platform management group can be shared. You can create separate management groups under the landing zone management group for each set of regulations that needs to be separated. You can assign the appropriate policies to each of the application management groups. The application landing zones share the management groups that are under the platform management group. The resources in the application management groups can also be separated by subscription or resource group.

This management group hierarchy is a simple and cost-effective design for isolating applications with conflicting regulations. However, in this design, the platform

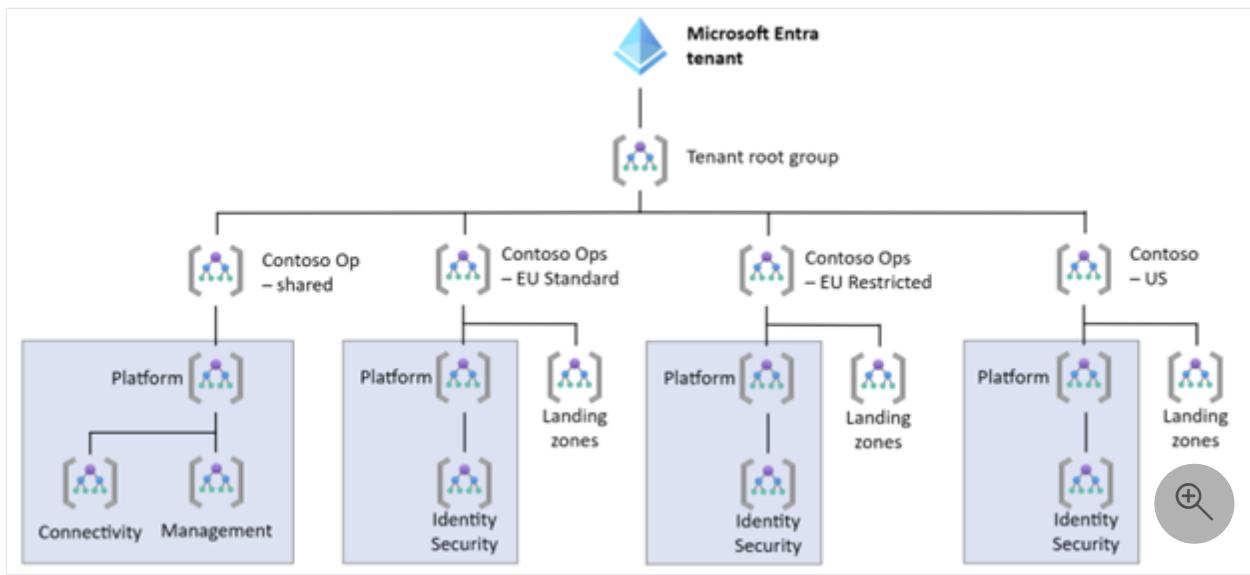
management groups for connectivity, identity/security, and management must share the same policy set. You might need different policy sets for each platform management group if regulation imposes restrictions on sharing connectivity infrastructure, identity services, key management services, or the infrastructure from which the whole environment is managed.



Isolate identity and security

If regulations prevent you from sharing the identity and key management infrastructure, you can divide the platform management group. Keep the management groups for connectivity and management in the shared platform management group and have an identity and security management group that's associated with each set of regulations.

This management group hierarchy is significantly more complex than a fully shared platform management group because you have to partially replicate the platform management group. To limit the complexity, you can deploy the full hierarchy for each of the regulation sets and the shared environment, and ignore or delete the superfluous management groups.

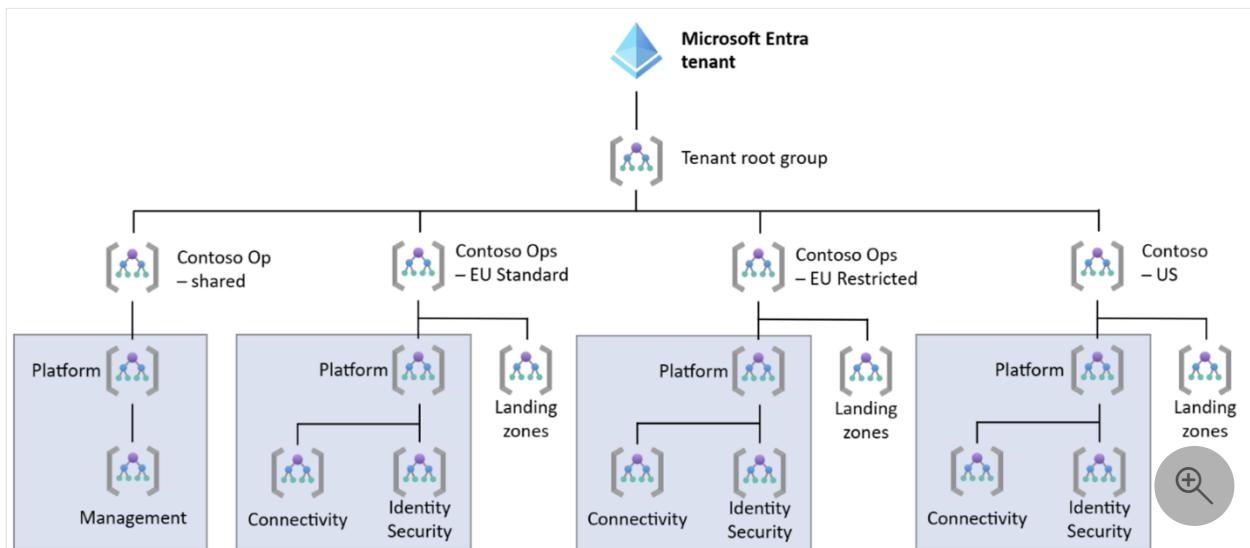


Isolate connectivity

Many regulations have requirements related to processing and storing data in a certain geographic location, with few requirements around how users connect to applications. For those regulations, you can share the connectivity management as shown in the previous architecture. There might not be any regulations that require you to duplicate infrastructure in multiple regions, but you might need to for latency purposes. The assigned policies need to support duplicating infrastructure in multiple regions.

When regulations have conflicting connectivity requirements, you can create a connectivity management group that's associated with each set of regulations. This structure is similar to the previous architecture that associates identity and security management groups with each set of regulations.

If regulations conflict for connectivity and also identity and security, you can use the following design.



Next steps

- Azure landing zones and multiple Microsoft Entra tenants
 - ISV considerations for Azure landing zones
 - Microsoft Cloud Adoption Framework for Azure
 - Microsoft Entra ID and data residency
 - Overview of the security pillar
 - Recommendations for identity and access management
 - Tailor the Azure landing zone architecture to meet requirements
-

Feedback

Was this page helpful?

 Yes

 No

Tailor the Azure landing zone architecture to meet requirements

Article • 11/03/2023

As part of the Azure landing zone guidance, several reference [implementation options](#) are available:

- Azure landing zone with Azure Virtual WAN
- Azure landing zone with traditional hub and spoke
- Azure landing zone foundation
- Azure landing zone for small enterprises

These options can help your organization get started quickly by using configurations that deliver the Azure landing zone conceptual architecture and best practices in the design areas.

The reference implementations are based on the best practices and learnings of Microsoft teams from engagements with customers and partners. This knowledge represents the "80" side of the 80/20 rule. The various implementations take positions on technical decisions that are part of the architecture design process.

Because not all use cases are the same, not all organizations can use an implementation approach in the exact way it was intended. You need to understand the considerations when a requirement for tailoring is identified.

What is a landing zone archetype in Azure landing zones?

A *landing zone archetype* describes what needs to be true to ensure a landing zone (Azure subscription) meets the expected environment and compliance requirements at a specific scope. Examples include:

- Azure Policy assignments.
- Role-based access control (RBAC) assignments.
- Centrally managed resources such as networking.

Consider each management group in the resource hierarchy as contributing to the final landing zone archetype output because of the way policy inheritance works in Azure. Think about what's applied at the upper levels in the resource hierarchy when you design the lower levels.

There's a close relationship between management groups and landing zone archetypes, but a management group alone isn't a landing zone archetype. Instead, it forms part of the framework that's used to implement each of the landing zone archetypes in your environment.

You can see this relationship in the Azure landing zone conceptual architecture. Policy assignments are created at the intermediate root management group, for example *Contoso*, for settings that must apply to all workloads. More policy assignments are created at lower levels of the hierarchy for more specific requirements.

Subscription placement within the management group hierarchy determines the resultant set of Azure Policy and access control (IAM) assignments that are inherited, applied, and enforced to that particular landing zone (Azure subscription).

More processes and tooling might be required to ensure a landing zone has the required centrally managed resources. Some examples include:

- Diagnostic settings to send activity log data to a Log Analytics workspace.
- Continuous export settings for Microsoft Defender for Cloud.
- Virtual network with managed IP address spaces for application workloads.
- Linking of virtual networks to a distributed denial of service (DDoS) Network Protection.

Note

In the Azure landing zone reference implementations, Azure policies with the `DeployIfNotExists` and `Modify` effects are used to achieve the deployment of some of the preceding resources. They follow the **policy-driven governance** design principle.

For more information, see [Adopt policy-driven guardrails](#).

Built-in archetypes for the Azure landing zone conceptual architecture

The conceptual architecture includes example landing zone archetypes for application workloads such as *corp* and *online*. These archetypes might apply to your organization and meet your requirements. You might want to make changes to these archetypes or create new ones. Your decision depends on your organization's needs and requirements.

Tip

To review the landing zone archetypes in the Azure landing zone accelerator, see [Management groups in the Azure landing zone accelerator](#).

You might also want to create changes elsewhere in the resource hierarchy. When you plan the hierarchy for your implementation of Azure landing zones for your organization, follow the guidelines in the [design areas](#).

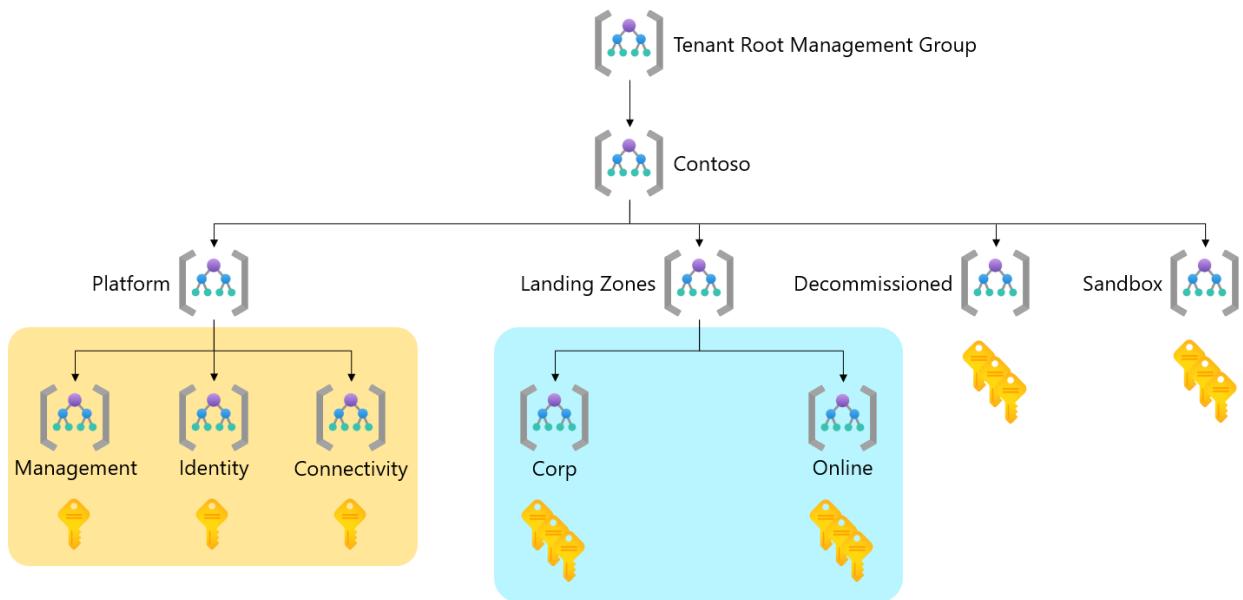
The following landing zone archetype examples from the conceptual architecture help you to understand their purpose and intended use:

Landing zone archetype (management group)	Purpose or use
Corp	The dedicated management group for corporate landing zones. This group is for workloads that require connectivity or hybrid connectivity with the corporate network via the hub in the connectivity subscription.
Online	The dedicated management group for online landing zones. This group is for workloads that might require direct internet inbound/outbound connectivity or for workloads that might not require a virtual network.
Sandbox	The dedicated management group for subscriptions that will only be used for testing and exploration by an organization. These subscriptions will be securely disconnected from the corporate and online landing zones. Sandboxes also have a less restrictive set of policies assigned to enable testing, exploration, and configuration of Azure services.

Scenarios where tailoring might be required

As mentioned, we provide common landing zone archetypes in [Azure landing zone conceptual architecture](#). They are *corp* and *online*. These archetypes aren't fixed and aren't the only permitted landing zone archetypes for application workloads. You might need to tailor landing zone archetypes to suit your needs and requirements.

Before you tailor landing zone archetypes, it's important to understand the concepts and also visualize the area of the hierarchy that we suggest you customize. The following diagram shows the default hierarchy of the Azure landing zone conceptual architecture.



Two areas of the hierarchy are highlighted. One is underneath **Landing Zones**, and the other is underneath **Platform**.

Tailor application landing zone archetypes

Notice the area highlighted in blue underneath the **Landing Zones** management group. It's the *most common and safest place* in the hierarchy to add more archetypes to meet new or more requirements that can't be added as more policy assignments to an existing archetype by using the existing hierarchy.

For example, you might have a new requirement to host a set of application workloads that need to meet payment card industry (PCI) compliance requirements. But this new requirement doesn't need to apply to all workloads across your entire estate.

There's a simple and safe way to meet this new requirement. Create a new management group called **PCI** underneath the **Landing Zones** management group in the hierarchy. You can assign more policies like the [Microsoft Defender for Cloud regulatory compliance policy initiative for PCI v3.2.1:2018](#) to the new **PCI** management group. This action forms a new archetype.

Now you can place new or move existing Azure subscriptions into the new **PCI** management group to make it inherit the required policies and form the new archetype.

Tip

You need to know what to consider and what happens when you move Azure subscriptions between management groups in relation to RBAC and Azure Policy.

For more information, see [Transition existing Azure environments to the Azure landing zone conceptual architecture](#).

Tailor platform landing zone archetypes

You might also want to tailor the area highlighted in orange underneath the **Platform** management group. The zones in this area are known as *platform landing zones*.

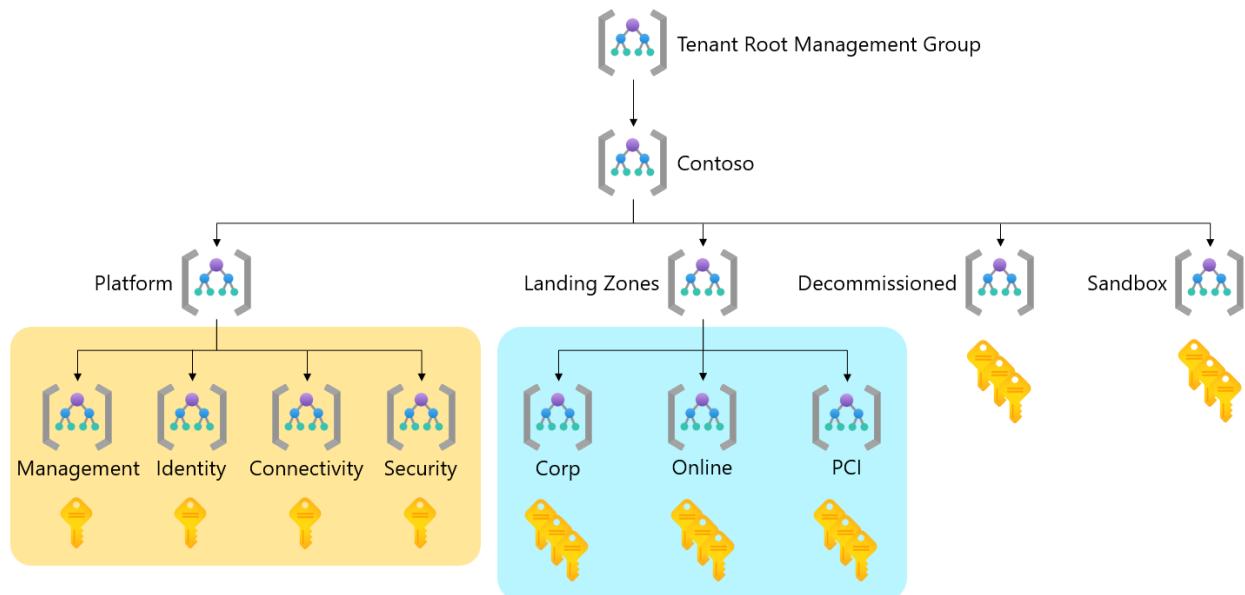
For example, you might have a dedicated SOC team that requires its own archetype to host its workloads. These workloads need to meet Azure Policy and RBAC assignment requirements different from those of the **Management** management group.

Create a new **Security** management group underneath the **Platform** management group in the hierarchy. You can assign the required Azure Policy and RBAC assignments to it.

Now you can place new or move existing Azure subscriptions into the new **Security** management group to make it inherit the required policies and form the new archetype.

Example of a tailored Azure landing zone hierarchy

The following diagram shows a tailored Azure landing zone hierarchy. It uses examples from the preceding diagram.



Points to consider

Consider the following points when you think about tailoring your implementation of Azure landing zone archetypes in the hierarchy:

- Tailoring the hierarchy isn't mandatory. The default archetypes and hierarchy we provide are suitable for most scenarios.
- Don't re-create your organizational hierarchy, teams, or departments in archetypes.
- Always try to build on the existing archetypes and hierarchy to meet new requirements.
- Only create new archetypes when they're truly needed.

For example, a new compliance requirement like PCI is required for only a subset of application workloads and doesn't need to apply to all workloads.

- Only create new archetypes in the highlighted areas shown in the preceding diagrams.
- Avoid going beyond a hierarchy depth of four layers to avoid complexity and unnecessary exclusions. Expand archetypes horizontally instead of vertically in the hierarchy.
- Don't create archetypes for environments like development, test, and production.

For more information, see [How do we handle dev/test/production workload landing zones in the Azure landing zones conceptual architecture?](#)

- If coming from a brownfield environment or are looking for an approach to host subscriptions in the Landing Zones Management Group with policies in an "audit only" enforcement mode, review [Scenario: Transition an environment by duplicating a landing zone management group](#)

Adopt policy-driven guardrails

Article • 05/31/2023

Before you use policies, you need to understand where they're used within the Azure landing zone reference implementations and why. This article will help you understand whether you want to prevent DeployIfNotExists (DINE) or Modify policies from making changes within your Azure environment.

Why use DINE and Modify policies?

DINE and Modify policies are part of the Azure landing zone reference implementations. They help you and your organization ensure your landing zones, which are also known as subscriptions, and the resources within them are compliant. These policies also remove the operational burden for platform and landing zone teams as your Azure environment scales.

For example, consider a scenario where a new landing zone subscription is provisioned and placed in the "corp" management group. DINE and Modify policies then take the following actions for the landing zone subscription:

- Enable Microsoft Defender for Cloud. Configure Defender for Cloud exports to the central Log Analytics workspace in the management subscription.
- Enable Defender for Cloud for the different supported offerings based on the policy parameters configured on the policy assignment.
- Configure the Azure Activity logs to be sent to the central Log Analytics workspace in the management subscription.
- Configure the diagnostic settings for all resources to be sent to the central Log Analytics workspace in the management subscription.
- Deploy the required Azure Monitor agents for virtual machines and Azure Virtual Machine Scale Sets, including Azure Arc connected servers. Connect them to the central Log Analytics workspace in the management subscription.

ⓘ Note

You can disable the preceding options at any time or during deployment of the Azure landing zone reference implementations.

The preceding list shows a subset of all the policies that are assigned as part of the Azure landing zone accelerator. For a full list of policies that can be assigned by the Azure landing zone reference implementation, see [Policies included in Azure landing zones reference implementations](#).

The Azure landing zones bicep repo [↗](#) is modular. The above default policies can be deployed with the ALZ Default Policy Assignments module [↗](#).

All assigned policies help you and the landing zone owners remain compliant. No actual workload resources are deployed via DINE or Modify policies. We don't recommend this either. For more information, see [Should we use Azure Policy to deploy workloads?](#). Only auxiliary or supporting resources or settings are deployed or configured by these DINE policies.

The Azure landing zones reference implementations use **DINE** Azure policies to help you achieve policy-driven governance within your Azure environment. But maybe you can't use DINE or Modify policies, or you aren't ready to enable this type of [Azure policy effect](#) because of:

- Regulatory compliance policies, standards, or law restrictions.
- Strict change control processes that require human approval for every action within your Azure environment.
- Lack of expertise, experience, and understanding of how to manage and use DINE policies.
- Organizational requirements that all workload resource configuration, including auxiliary resources, supporting resources, and settings, are defined in Infrastructure as Code (IaC) by the workload application teams.

If you fit into the preceding examples or similar scenarios, this article helps you understand how to adopt the [Azure landing zone conceptual architecture](#) and adhere to its [design principles](#). Although you won't use certain policies initially, you can choose to gradually enable them in the future. The goal is to help you achieve [policy-driven governance](#).

Important

Throughout this article, you'll see two possible values used for the enforcement mode terms:

- Disabled or DoNotEnforce
- Enabled or Default

The Azure portal uses Disabled and Enabled for the enforcement mode. Azure Resource Manager (ARM) templates and other API interfaces use DoNotEnforce and Default for the same options.

For more information, see [Enforcement mode](#).

If you're still certain that your organization can't use DINE or Modify policies, this article explains how to prevent (also known as disable) the policies from making automatic changes to your Azure environment.

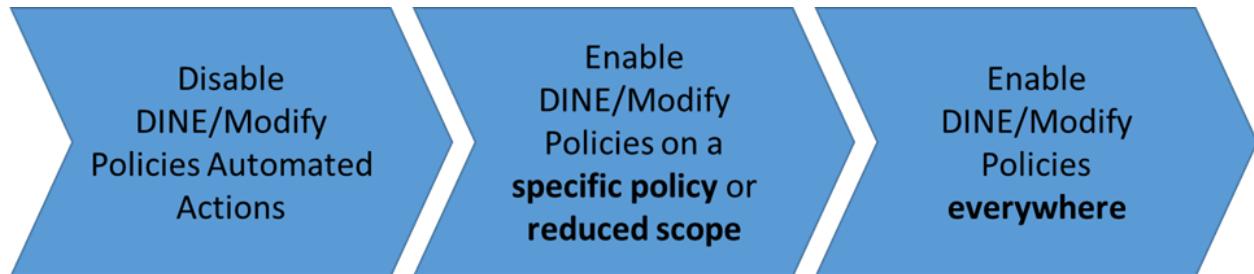
ⓘ Note

This operation isn't permanent. The policies can be reenabled at any time by a member of your platform team if you later decide to use DINE or Modify policies.

For more information, see [phase 2](#) and [phase 3](#).

Approach overview

The following diagram summarizes the suggested phased approach:



1. Set the [enforcement mode](#) to `DoNotEnforce` on policy assignments:

- By using this feature, you can modify the assignments' behavior to effectively become an audit-only policy without modifying the underlying policy definition.
- This approach also allows you to do manual remediation tasks on noncompliant resources by using [remediation tasks](#) if you want to.

2. Set the [enforcement mode](#) to `Default` on policy assignments to reenable DINE policy assignments' automatic remediation *on a reduced scope*:

- You can choose to use an entire environment, for example, the Sandbox management group.
- Or, you can use a noncritical workload subscription.

3. Set the [enforcement mode](#) to `Default` on policy assignments on the remaining DINE policies across the entire Azure environment.

Because of regulatory compliance restrictions, some customers can never move past phase 1. This isn't an issue and is supported to remain in this state, if necessary. Other

customers can progress to phases 2 and 3 to fully adopt DINE and Modify policies to assist with policy-driven governance for their Azure environment.

 **Note**

The scenario and approach outlined in this article isn't intended for or recommended for the majority of customers. Review the section [Why use DINE and Modify policies?](#) before you decide whether these policies are suitable and required for your environment.

Phase 1: Disable DINE and Modify policies automated actions

When you assign a policy, by default the [effect](#) defined in the policy definition will apply. We recommend that you leave the policy definition as is. For example, leave the policy assignment effect as `DeployIfNotExists`.

Instead of changing the policy definition or its effect, you can instead influence this behavior with minimal effort by using the feature on policy assignments.

Use the Azure portal to set the enforcement mode to Disabled

This screenshot shows how to use the Azure portal to set the enforcement mode to **Disabled** on a policy assignment. Disabled is also known as `DoNotEnforce`.

Deploy Azure Security Center configuration

Edit Initiative Assignment

Basics Parameters Remediation Non-compliance messages Review + save

Scope
Scope [Learn more about setting the scope](#)

jt-102021 ...

Exclusions
 Optionally select resources to exclude from the policy assignment. ...

Basics

Policy definition
[Deploy Azure Security Center configuration](#)

Assignment name * ⓘ
[Deploy Azure Security Center configuration](#)

Assignment ID
[/providers/Microsoft.Management/managementGroups/jt-102021/providers/Microsoft.Authorization/policyAssignme... ↗](#)

Description
Deploy ASC configuration for Azure Defender and Security Contacts

Policy enforcement ⓘ
 Enabled Disabled
 (highlighted)

Use the ARM template to set the enforcement mode to DoNotEnforce

This code example shows how to use an ARM template to set `enforcementMode` to `DoNotEnforce` on a policy assignment. `DoNotEnforce` is also known as `Disabled`.

JSON

```
{
  "type": "Microsoft.Authorization/policyAssignments",
  "apiVersion": "2019-09-01",
  "name": "PolicyAssignmentName",
  "location": "[deployment().location]",
  "properties": {
    "description": "PolicyAssignmentDescription",
    "policyDefinitionId": "[parameters('policyDefinitionId')]",
    "enforcementMode": "DoNotEnforce"
    ... // other properties removed for display purposes
  }
}
```

By using the `enforcement mode`, you can see the effect of a policy on existing resources without initiating it or triggering entries in the Azure Activity log. This scenario is commonly referred to as "What If" and aligns to safe deployment practices.

Even when the [enforcement mode](#) is set to `DoNotEnforce`, [remediation tasks](#) can be triggered manually. You can remediate specific noncompliant resources. You can also see what the DINE or Modify policy would have done if the enforcement mode were set to `Default`.

Important

When the enforcement mode is set to `DoNotEnforce`, entries in the Azure Activity log aren't generated. Consider this factor if you want to be notified when a noncompliant resource is created.

Stay in the phase 1 state permanently

As mentioned in the [Approach overview](#) section, some customers might need to remain in [phase 1](#) for a long period or even permanently because of their requirements. This state is valid, and customers can remain in it for any length of time.

Perhaps you need to stay in this state permanently or for a long period, like years. If so, it might be better for you to adopt the [AuditIfNotExists](#) (AINE) policy effect and associated definitions and set the enforcement mode back to `Default`.

Note

By changing to using an AINE policy and setting the enforcement mode to `Default`, you still achieve the same goal of disabling DINE.

When you change from DINE to AINE and set the enforcement mode back to `Default` as a long-term or permanent approach for phase 1, you'll gain back the Azure Activity log entries for policy compliance statuses. You can build automation workflows from these log entries in your overall platform management operations.

You'll lose the capability to do manual remediation tasks. Unlike DINE policies, AINE policies don't perform any deployments, either automated or manual.

Remember to update the policy definition to accept and allow the `AuditIfNotExists` policy assignment effect.

The following table summarizes the options and implications for the different types of policy effects and enforcement mode combinations:

Policy effect	Enforcement mode	Activity log entry	Remediation action
DINE	Enabled or Default	Yes	Platform-triggered remediation at scale after creation or resource update. Manual creation of a remediation task required if dependent resource is modified or preexisting prior to the policy assignment.
DINE	Disabled or DoNotEnforce	No	Manual creation of a remediation task required.
Modify	Enabled or Default	Yes	Automatic remediation during creation or update.
Modify	Disabled or DoNotEnforce	No	Manual creation of a remediation task required.
Deny	Enabled or Default	Yes	Creation or update denied.
Deny	Disabled or DoNotEnforce	No	Creation or update allowed. Manual remediation required.
Audit/AINE	Enabled or Default	Yes	Manual remediation required.
Audit/AINE	Disabled or DoNotEnforce	No	Manual remediation required.

Note

Review the guidance in [Reacting to Azure Policy state change events](#) to understand if using the Azure Event Grid integration with Azure Policy provides a suitable approach if you plan to build your own automation based on policy state events.

Phase 2: Enable DINE and Modify policies on a specific policy or reduced scope

In this phase, you'll learn how to set the enforcement mode to `Default` on policy assignments.

After you've completed [phase 1](#), you decide that you want to test and try out the full automation capabilities of DINE and Modify policies on a specific policy or on a reduced

scope. You want to use the Sandbox management group or a nonproduction workload subscription.

To do this procedure, first you need to identify the policy or reduced scope that will be used to test and try the DINE and Modify policies' full automation capabilities.

ⓘ Note

You might want to review and implement a **testing approach for an enterprise-scale** platform. In this way, you can test policies and other platform changes in a separated management group hierarchy within the same tenant.

This approach is also known as a "canary" deployment.

Some suggested examples of scopes and policies are shown in the following table:

When you want to...	...choose from these scopes	Example policies to use
<ul style="list-style-type: none">- Test the DINE/Modify automated remediation capabilities.- Verify how your complete deployment processes and CI/CD pipelines, including tests, might be affected.- Verify how your workload might be affected.	<ul style="list-style-type: none">- Sandbox subscription- Sandbox management group- Nonproduction workload landing zone subscription- Enterprise-scale "canary" environment	<ul style="list-style-type: none">- Configure Azure Activity logs to stream to a specified Log Analytics workspace.- Deploy Defender for Cloud configuration.- Enable Azure Monitor for VMs or Virtual Machine Scale Sets.- Deploy diagnostic settings to Azure services.- Potentially only enable for specific services within the initiative.

You might also decide to use a manual remediation task on a limited scope or set of resources to test how these policies will affect your environment. For more information on how to create a remediation task, see the Azure Policy documentation [Create a remediation task](#).

After you've identified a policy, or policies, and the reduced scope to assign them, the next step is to assign the policy and set the enforcement mode to **Default**. Leave the policy effect, for example, **DeployIfNotExists** or **Modify**, as is on the reduced scope you selected.

Use the Azure portal to set the enforcement mode to Enabled

This screenshot shows how to use the Azure portal to set the enforcement mode to **Enabled** on a policy assignment. Enabled is also known as Default.

Home > Policy > Deploy Azure Security Center configuration >

Deploy Azure Security Center configuration

Edit Initiative Assignment

Basics Parameters Remediation Non-compliance messages Review + save

Scope

Scope [Learn more about setting the scope](#)

jt-102021

Exclusions

Optionally select resources to exclude from the policy assignment.

Basics

Policy definition

Deploy Azure Security Center configuration

Assignment name * ⓘ

Deploy Azure Security Center configuration

Assignment ID

/providers/Microsoft.Management/managementGroups/jt-102021/providers/Microsoft.Authorization/policyAssignme... [Copy](#)

Description

Deploy ASC configuration for Azure Defender and Security Contacts

Policy enforcement ⓘ

Enabled **Disabled**

Use an ARM template to set the enforcement mode to Default

This code example shows how to use an ARM template to set `enforcementMode` to `Default` on a policy assignment. `Default` is also known as `Enabled`.

JSON

```
{
  "type": "Microsoft.Authorization/policyAssignments",
  "apiVersion": "2019-09-01",
  "name": "PolicyAssignmentName",
  "location": "[deployment().location]",
  "properties": {
    "description": "PolicyAssignmentDescription",
    "policyDefinitionId": "[parameters('policyDefinitionId')]",
    "enforcementMode": "Default"
    ... // other properties removed for display purposes
  }
}
```

Testing

The last step in this phase is to do the required testing. You want to verify whether and how DINE or Modify policies might have affected and made changes to your workloads, code, tools, and processes.

Perform multiple tests to capture the entire lifecycle of your workload. You want to ensure you fully understand if and how DINE or Modify policies made changes.

Some examples of testing are:

- Initial deployment of workload.
- Code/Application deployment onto workload.
- Day 2 operations and management of workload.
- Decommissioning of workload.

Phase 3: Enable DINE and Modify policies everywhere

In this phase, you'll learn how to set the enforcement mode to `Default` on policy assignments.

We assume that your [testing](#) at the end of [phase 2](#) passed successfully. Or, maybe you're satisfied that you now understand how DINE or Modify policies interact with your workload. Now you can expand the use of DINE and Modify policies across the rest of your Azure environment.

To proceed, you follow steps that are similar to the steps in [phase 2](#). This time, you set the enforcement mode to `Default` on all DINE and Modify policy assignments across your entire Azure environment.

Here's a high-level overview of the steps you do in this phase:

- Remove assignments used specifically for [testing during phase 2](#).
- Go through each DINE and Modify policy assignment in your Azure environment and set the enforcement mode to `Default`. This process is shown in the examples in phase 2.
- Create remediation tasks for existing resources that are noncompliant by following the guidance in [Create a remediation task](#). New resources will automatically be remediated if they match the policy rules and existence conditions.

Even though in phase 3 we recommend that you set the enforcement mode to `Default` for all DINE and Modify policies in your Azure environment, this choice is still optional. You can make this choice on a per-policy basis to suit your needs and requirements.

Develop your naming and tagging strategy for Azure resources

Article • 03/22/2023

Organize your cloud assets to support governance, operational management, and accounting requirements. Well-defined naming and metadata tagging conventions help to quickly locate and manage resources. These conventions also help associate cloud usage costs with business teams via chargeback and showback accounting mechanisms.

Define your naming and tagging strategy as early as possible. Use the following links to help you define and implement your strategy:

- [Define your naming convention](#)
- [Recommended abbreviations for Azure resource types](#)
- [Define your tagging strategy](#)
- [Resource naming and tagging decision guide](#)
- [Naming rules and restrictions for Azure resources](#)

ⓘ Note

Every business has its own organizational and management requirements. These recommendations help start a discussion with your cloud adoption teams. As the discussion proceeds, use the tools below to document the naming and tagging decisions you make when aligning these recommendations to your specific business needs.

Download the [Azure Naming Tool](#) to create an organizational naming reference and name generator.

Download the [naming and tagging conventions tracking template](#).

Purpose of naming and tagging

Accurately representing and naming your resources is essential for security purposes. If you come upon a security incident, it's critical to quickly identify affected systems, what functions those systems support, and the potential business impact. Security services such as [Microsoft Defender for Cloud](#) and [Microsoft Sentinel](#) reference resources and their associated logging and alert information by resource name.

Azure defines [naming rules and restrictions for Azure resources](#). This guidance provides you with detailed recommendations to support enterprise cloud adoption efforts.

Changing resource names can be difficult. Establish a comprehensive naming convention before you begin any large cloud deployment.

Naming and tagging strategy

A naming and tagging strategy includes business and operational details as components of resource names and metadata tags:

- The business side of this strategy ensures that resource names and tags include the organizational information you need to identify the teams. Use a resource along with the business owners who are responsible for resource costs.
- The operational side ensures that names and tags include necessary information. IT teams use this information to identify the workload, application, environment, criticality, and other information useful for managing resources.

Next steps

Learn about the considerations for defining your naming convention of your Azure resources and assets, and review example names for resources and assets in Azure.

[Name your Azure resources and assets](#)

Define your naming convention

Article • 12/13/2023

A good name for a resource helps you to quickly identify its type, its associated workload, its environment, and the Azure region where it runs. To do so, names should follow a consistent format—a *naming convention*—that is composed of important information about each resource. The information in the names ideally includes whatever you need to identify specific instances of resources. For example, a public IP address (PIP) for a production SharePoint workload in the West US region might be `pip-sharepoint-prod-westus-001`.

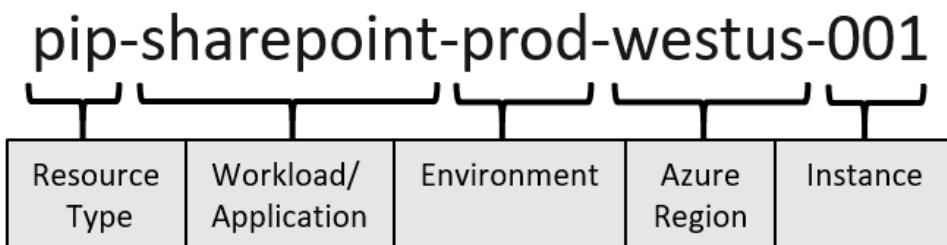


Diagram 1: Components of an Azure resource name.

Recommended naming components

When you construct your naming convention, identify the key pieces of information that you want to capture in a resource name. Different information is relevant for different resource types, and not all established naming components can be used for each resource type. Establish a standard naming convention for your environment that is easy to follow, concise, and useful for recognizing information that's relevant to the deployed resource.

The following list provides examples of naming components that are useful when you construct resource names:

Expand table

Naming component	Description
Organization	Top-level name of the organization, normally utilized as the top management group or, in smaller organizations, part of the naming convention. Example: <code>contoso</code>
Business unit or department	Top-level division of your company that owns the subscription or the workload that the resource belongs to. In smaller organizations, this

Naming component	Description
	component might represent a single corporate, top-level organizational element. Examples: <code>fin</code> , <code>mktg</code> , <code>product</code> , <code>it</code> , <code>corp</code>
Resource type	An abbreviation that represents the type of Azure resource or asset. This component is often a prefix or suffix in the name. For more information, see Recommended abbreviations for Azure resource types . Examples: <code>rg</code> , <code>vm</code>
Project, application, or service name	Name of a project, application, or service that the resource is a part of. Examples: <code>navigator</code> , <code>emissions</code> , <code>sharepoint</code> , <code>hadoop</code>
Environment	The stage of the development lifecycle for the workload that the resource supports. Examples: <code>prod</code> , <code>dev</code> , <code>qa</code> , <code>stage</code> , <code>test</code>
Location	The region or cloud provider where the resource is deployed. Examples: <code>westus</code> , <code>eastus2</code> , <code>westeu</code> , <code>usva</code> , <code>ustx</code>
VM role	Identifier of the purpose of the VM. Examples: <code>db</code> (database), <code>ws</code> (web server), <code>ps</code> (print server)
Instance	The instance count for a specific resource, to differentiate it from other resources that have the same naming convention and naming components. Examples, <code>01</code> , <code>001</code>

ⓘ Note

Although virtual machine (VM) names in Azure can be longer than the allowed NetBIOS name of the VM, we recommend that you keep them consistent. For more information and for other restrictions, see [Computer names](#).

Naming considerations

In addition to defining the naming components, you must also consider the order in which the naming components are listed and what type of delimiters (if any) should appear between components. Also take into account the different naming rules that are associated with resources types.

Scope

All Azure resource types have a scope that defines the level of that resource. Also, A resource must have a unique name within its scope.

For example, a virtual network has the scope of a resource group, which means that there can be only one network named `vnet-prod-westus-001` in a specific resource group. Other resource groups can also have virtual networks named `vnet-prod-westus-001`, but each resource group can have only one with that name. Subnets are scoped to virtual networks, so each subnet within a virtual network must have a distinct name.

Some resource names have a global scope, such as a name for a Platform as a Service (PaaS) that has a public endpoint or a virtual machine DNS label. A resource in a global scope must have a name that's unique across the entire Azure platform.

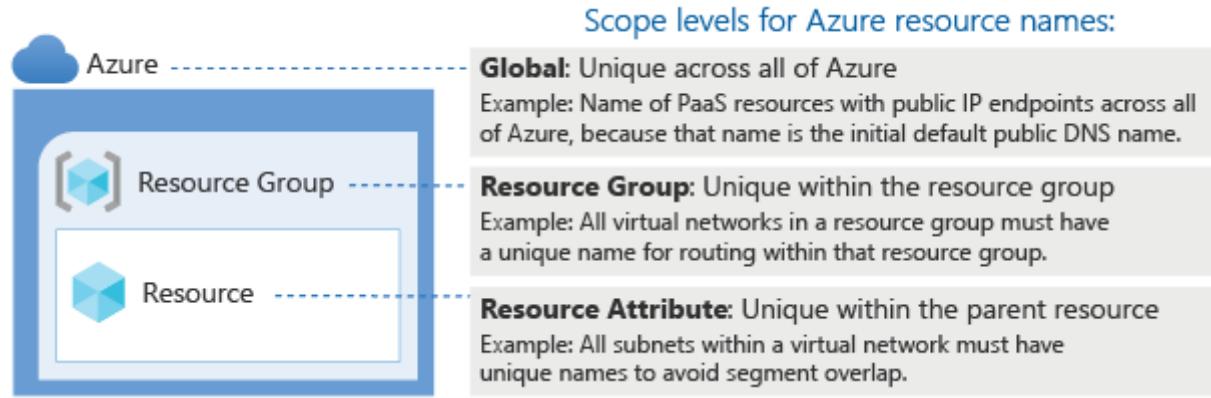


Diagram 2: Scope levels for Azure resource names.

Azure naming rules

Azure naming rules vary depending on the resource type. When you define a naming convention, it's important to understand Azure naming rules for the resource type to avoid confusion and delay deployments.

For example, resource names have length limits. We recommend that you keep the length of naming components short to prevent exceeding resource name length limits.

! Note

Balancing the context of a name with its scope and length limit is important when you develop your naming conventions. For more information, see [Naming rules and restrictions for Azure resources](#).

When you construct your naming convention, identify the key pieces of information that you want to reflect in a resource name. Different information is relevant for different resource types. The following list provides examples of information that are useful when you construct resource names.

You can abbreviate resource names and naming components as a strategy to reduce the length and complexity of resource names. Shortening names can be useful for any of the naming components, but it's especially important to help you keep resource names within name length limits. For example, a VM name in Azure can be longer than the OS naming restrictions. Keeping Azure VM names shorter than the naming restrictions of the OS helps create consistency, improve communication when discussing resources, and reduce confusion when you're working in the Azure portal while being signed in to the VM itself.

 Expand table

Naming component	Description
Resource type	An abbreviation that represents the type of Azure resource or asset. This component is often used as a prefix or suffix in the name. For more information, see Recommended abbreviations for Azure resource types . Examples: <code>rg</code> , <code>vm</code>
Business unit	Top-level division of your company that owns the subscription or workload the resource belongs to. In smaller organizations, this component might represent a single corporate top-level organizational element. Examples: <code>fin</code> , <code>mktg</code> , <code>product</code> , <code>it</code> , <code>corp</code>
Application or service name	Name of the application, workload, or service that the resource is a part of. Examples: <code>navigator</code> , <code>emissions</code> , <code>sharepoint</code> , <code>hadoop</code>
Subscription purpose	Summary description of the purpose of the subscription that contains the resource. Often broken down by environment or specific workloads. Examples: <code>prod</code> , <code>shared</code> , <code>client</code>
Environment	The stage of the development lifecycle for the workload that the resource supports. Examples: <code>prod</code> , <code>dev</code> , <code>qa</code> , <code>stage</code> , <code>test</code>
Region	The Azure region where the resource is deployed. Examples: <code>westus</code> , <code>eastus2</code> , <code>westeu</code> , <code>usva</code> , <code>ustx</code>

 Note

When you're ready to name your resources and assets, review [Recommended abbreviations for Azure resource types](#).

The following section provides example names for common Azure resource types in an enterprise cloud deployment.

Note

Some of these example names use a three-digit padding scheme (###), such as `mktg-prod-001`.

Padding improves readability and sorting of assets when those assets are managed in a configuration management database (CMDB), IT Asset Management tool, or traditional accounting tools. When the deployed asset is managed centrally as part of a larger inventory or portfolio of IT assets, the padding approach aligns with interfaces those systems use to manage inventory naming.

Unfortunately, the traditional asset padding approach can prove problematic in infrastructure-as-code approaches that might iterate through assets based on a non-padded number. This approach is common during deployment or automated configuration management tasks. Those scripts would have to routinely strip the padding and convert the padded number to a real number, which slows script development and run time.

Choose an approach that's suitable for your organization. The padding shown here illustrates the importance of using a consistent approach to inventory numbering, rather than showing which approach is superior. Before choosing a numbering scheme, with or without padding, evaluate what will affect long-term operations more: CMDB and asset management solutions or code-based inventory management. Then, consistently follow the padding option that best fits your operational needs.

The following section provides some example names for common Azure resource types in an enterprise cloud deployment. For more examples, see the [Azure Naming Tool](#) and the [Naming and tagging tracking template](#).

Note

The following examples are intended to provide visualization of a naming convention, but actual conventions vary by organization.

Example names: General

 Expand table

Asset type	Scope	Format and examples
Management group	Business unit and/or environment	<p><i>mg-<business unit>[-<environment>]</i></p> <ul style="list-style-type: none"> • <code>mg-mktg</code> • <code>mg-hr</code> • <code>mg-corp-prod</code> • <code>mg-fin-client</code>
Subscription	Account / enterprise agreement	<p><i><business unit>-<subscription purpose>-<###></i></p> <ul style="list-style-type: none"> • <code>mktg-prod-001</code> • <code>corp-shared-001</code> • <code>fin-client-001</code>
Resource group	Subscription	<p><i>rg-<app or service name>-<subscription purpose>-<###></i></p> <ul style="list-style-type: none"> • <code>rg-mktgsharepoint-prod-001</code> • <code>rg-acctlookupsvc-shared-001</code> • <code>rg-ad-dir-services-shared-001</code>
API management service instance	Global	<p><i>apim-<app or service name></i></p> <p><code>apim-navigator-prod</code></p>
Managed identity	Resource group	<p><i>id-<app or service name>-<environment>-<region name>-<###></i></p> <ul style="list-style-type: none"> • <code>id-appcn-keda-prod-eastus2-001</code>

Example names: Networking

[\[+\] Expand table](#)

Asset type	Scope	Format and examples
Virtual network	Resource group	<p><i>vnet-<subscription purpose>-<region>-<###></i></p> <ul style="list-style-type: none"> • <code>vnet-shared-eastus2-001</code> • <code>vnet-prod-westus-001</code> • <code>vnet-client-eastus2-001</code>
Subnet	Virtual network	<p><i>snet-<subscription purpose>-<region>-<###></i></p> <ul style="list-style-type: none"> • <code>snet-shared-eastus2-001</code> • <code>snet-prod-westus-001</code>

Asset type	Scope	Format and examples
		<ul style="list-style-type: none"> • snet-client-eastus2-001
Network interface (NIC)	Resource group	<p><i>nic-<##>-<vm name>-<subscription purpose>-<###></i></p> <ul style="list-style-type: none"> • nic-01-dc1-shared-001 • nic-02-vmhadoop1-prod-001 • nic-02-vmtest1-client-001
Public IP address	Resource group	<p><i>pip-<vm name or app name>-<environment>-<region>-<###></i></p> <ul style="list-style-type: none"> • pip-dc1-shared-eastus2-001 • pip-hadoop-prod-westus-001
Load balancer (external)	Resource group	<p><i>lbe-<app name or role>-<environment>-<###></i></p> <ul style="list-style-type: none"> • lbe-navigator-prod-001 • lbe-sharepoint-dev-001
Network security group (NSG)	Subnet or NIC	<p><i>nsg-<policy name or app name>-<###></i></p> <ul style="list-style-type: none"> • nsg-weballow-001 • nsg-rdpallow-001 • nsg-sqlallow-001 • nsg-dnsblocked-001
Local network gateway	Virtual gateway	<p><i>lgw-<subscription purpose>-<region>-<###></i></p> <ul style="list-style-type: none"> • lgw-shared-eastus2-001 • lgw-prod-westus-001 • lgw-client-eastus2-001
Virtual network gateway	Virtual network	<p><i>vgw-<subscription purpose>-<region>-<###></i></p> <ul style="list-style-type: none"> • vgw-shared-eastus2-001 • vgw-prod-westus-001 • vgw-client-eastus2-001
VPN connection	Resource group	<p><i>vcn-<subscription1 purpose>>-<region1>-to-<subscription2 purpose>>-<region2>-</i></p> <ul style="list-style-type: none"> • vcn-shared-eastus2-to-shared-westus • vcn-prod-eastus2-to-prod-westus
Route table	Resource group	<p><i>rt-<route table name></i></p> <ul style="list-style-type: none"> • rt-navigator • rt-sharepoint

Asset type	Scope	Format and examples
DNS label	Global	<p><i><DNS A record for VM>. <region>.cloudapp.azure.com</i></p> <ul style="list-style-type: none"> dc1.westus.cloudapp.azure.com web1.eastus2.cloudapp.azure.com

Example names: Compute and Web

[Expand table](#)

Asset type	Scope	Format and examples
Virtual machine	Resource group	<p><i>vm-<vm role>-<environment>-<###></i></p> <ul style="list-style-type: none"> vm-sql-test-001 vm-hadoop-prod-001
Web app	Global	<p><i>app-<project, app or service>-<environment>-<###>.azurewebsites.net</i></p> <ul style="list-style-type: none"> app-navigator-prod-001.azurewebsites.net app-accountlookup-dev-001.azurewebsites.net
Function app	Global	<p><i>func-<project, app or service>-<environment>-<###>.azurewebsites.net</i></p> <ul style="list-style-type: none"> func-navigator-prod-001.azurewebsites.net func-accountlookup-dev-001.azurewebsites.net

Example names: Databases

[Expand table](#)

Asset type	Scope	Format and examples
Azure SQL database	Azure SQL Server	<p><i>sqldb-<project, app or service>-<environment></i></p> <ul style="list-style-type: none"> sqldb-users-prod sqldb-users-dev
Azure Cosmos DB database	Global	<p><i>cosmos-<project, app or service>-<environment></i></p> <ul style="list-style-type: none"> cosmos-navigator-prod cosmos-emissions-dev

Asset type	Scope	Format and examples
Azure Cache for Redis instance	Global	<p><i>redis-<project, app or service>-<environment></i></p> <ul style="list-style-type: none"> • <code>redis-navigator-prod</code> • <code>redis-emissions-dev</code>

Example names: Storage

[Expand table](#)

Asset type	Scope	Format and examples
Storage account (general use)	Global	<p><i>st<project, app or service><###></i></p> <ul style="list-style-type: none"> • <code>stnavigatordata001</code> • <code>stemissionsoutput001</code>
Azure StorSimple	Global	<p><i>ssimp<project, app or service><environment></i></p> <ul style="list-style-type: none"> • <code>ssimpnavigatorprod</code> • <code>ssimpemissionsdev</code>
Azure Container Registry	Global	<p><i>cr<project, app or service><environment><###></i></p> <ul style="list-style-type: none"> • <code>crnavigatorprod001</code>

Example names: AI and machine learning

[Expand table](#)

Asset type	Scope	Format and examples
Azure AI Search	Global	<p><i>srch-<project, app or service>-<environment></i></p> <ul style="list-style-type: none"> • <code>srch-navigator-prod</code> • <code>srch-emissions-dev</code>
Azure OpenAI Service	Resource group	<p><i>oai-<project, app or service>-<environment></i></p> <ul style="list-style-type: none"> • <code>oai-navigator-prod</code> • <code>oai-emissions-dev</code>

Asset type	Scope	Format and examples
Azure Machine Learning workspace	Resource group	<p><i>mlw-<project, app or service>-<environment></i></p> <ul style="list-style-type: none"> • <code>mlw-navigator-prod</code> • <code>mlw-emissions-dev</code>

Example names: Analytics and IoT

[Expand table](#)

Asset type	Scope	Format and examples
Azure Analysis Services	Global	<p><i>as<app name><environment></i></p> <ul style="list-style-type: none"> • <code>asnavigatordprod</code> • <code>asemissionsdev</code>
Azure Data Factory	Global	<p><i>adf-<project, app or service>-<environment></i></p> <ul style="list-style-type: none"> • <code>adf-navigator-prod</code> • <code>adf-emissions-dev</code>
Azure Synapse Analytics workspaces	Resource group	<p><i>synw-<project, app or service>-<environment></i></p> <ul style="list-style-type: none"> • <code>synw-navigator-prod</code> • <code>synw-emissions-dev</code>
Data Lake Storage account	Global	<p><i>dls<project, app or service><environment></i></p> <ul style="list-style-type: none"> • <code>dlsnavigatorprod</code> • <code>dlsemissionsdev</code>
IoT hub	Global	<p><i>iot-<project, app or service>-<environment></i></p> <ul style="list-style-type: none"> • <code>iot-navigator-prod</code> • <code>iot-emissions-dev</code>

Example names: Integration

[Expand table](#)

Asset type	Scope	Format and Examples
Service Bus namespace	Global	<p><i>sbns-<project, app or service>-<environment>.servicebus.windows.net</i></p> <ul style="list-style-type: none"> • <code>sbns-navigator-prod.servicebus.windows.net</code> • <code>sbns-emissions-dev.servicebus.windows.net</code>
Service Bus queue	Service Bus	<p><i>sbq-<query descriptor></i></p> <ul style="list-style-type: none"> • <code>sbq-messagequery</code>
Service Bus topic	Service Bus	<p><i>sbt-<query descriptor></i></p> <ul style="list-style-type: none"> • <code>sbt-messagequery</code>

Abbreviation recommendations for Azure resources

Article • 05/07/2024

This page gives you abbreviation examples for many of the resources in Azure. The following table has *abbreviations* mapped to *resource* and *resource provider namespace*.

Azure Naming Tool: You can use the Azure Naming Tool to standardize and automate your naming process. For more information, see [Azure Naming Tool Overview](#).

AI + machine learning

[+] Expand table

Resource	Resource provider namespace	Abbreviation
AI Search	Microsoft.Search/searchServices	srch
Azure AI services multi-service account	Microsoft.CognitiveServices/accounts (kind: CognitiveServices)	aisa
Azure AI Video Indexer	Microsoft.VideoIndexer/accounts	avi
Azure Machine Learning workspace	Microsoft.MachineLearningServices/workspaces	mlw
Azure OpenAI Service	Microsoft.CognitiveServices/accounts (kind: OpenAI)	oai
Bot service	Microsoft.BotService/botServices (kind: azurebot)	bot
Computer vision	Microsoft.CognitiveServices/accounts (kind: ComputerVision)	cv
Content moderator	Microsoft.CognitiveServices/accounts (kind: ContentModerator)	cm
Content safety	Microsoft.CognitiveServices/accounts (kind: ContentSafety)	cs
Custom vision (prediction)	Microsoft.CognitiveServices/accounts (kind: CustomVision.Prediction)	cstv
Custom vision (training)	Microsoft.CognitiveServices/accounts (kind: CustomVision.Training)	cstvt

Resource	Resource provider namespace	Abbreviation
Document intelligence	Microsoft.CognitiveServices/accounts (kind: FormRecognizer)	di
Face API	Microsoft.CognitiveServices/accounts (kind: Face)	face
Health Insights	Microsoft.CognitiveServices/accounts (kind: HealthInsights)	hi
Immersive reader	Microsoft.CognitiveServices/accounts (kind: ImmersiveReader)	ir
Language service	Microsoft.CognitiveServices/accounts (kind: TextAnalytics)	lang
Speech service	Microsoft.CognitiveServices/accounts (kind: SpeechServices)	spch
Translator	Microsoft.CognitiveServices/accounts (kind: TextTranslation)	trs1

Analytics and IoT

[Expand table](#)

Resource	Resource provider namespace	Abbreviation
Azure Analysis Services server	Microsoft.AnalysisServices/servers	as
Azure Databricks workspace	Microsoft.Databricks/workspaces	dbw
Azure Data Explorer cluster	Microsoft.Kusto/clusters	dec
Azure Data Explorer cluster database	Microsoft.Kusto/clusters/databases	dedb
Azure Data Factory	Microsoft.DataFactory/factories	adf
Azure Digital Twin instance	Microsoft.DigitalTwins/digitalTwinsInstances	dt
Azure Stream Analytics	Microsoft.StreamAnalytics/cluster	asa
Azure Synapse Analytics private link hub	Microsoft.Synapse/privateLinkHubs	synplh

Resource	Resource provider namespace	Abbreviation
Azure Synapse Analytics SQL Dedicated Pool	Microsoft.Synapse/workspaces/sqlPools	syndp
Azure Synapse Analytics Spark Pool	Microsoft.Synapse/workspaces/bigDataPools	synsp
Azure Synapse Analytics workspaces	Microsoft.Synapse/workspaces	synw
Data Lake Store account	Microsoft.DataLakeStore/accounts	dls
Data Lake Analytics account	Microsoft.DataLakeAnalytics/accounts	dla
Event Hubs namespace	Microsoft.EventHub/namespaces	evhns
Event hub	Microsoft.EventHub/namespaces/eventHubs	evh
Event Grid domain	Microsoft.EventGrid/domains	evgd
Event Grid subscriptions	Microsoft.EventGrid/eventSubscriptions	evgs
Event Grid topic	Microsoft.EventGrid/domains/topics	evgt
Event Grid system topic	Microsoft.EventGrid/systemTopics	egst
HDInsight - Hadoop cluster	Microsoft.HDInsight/clusters	hadoop
HDInsight - HBase cluster	Microsoft.HDInsight/clusters	hbase
HDInsight - Kafka cluster	Microsoft.HDInsight/clusters	kafka
HDInsight - Spark cluster	Microsoft.HDInsight/clusters	spark
HDInsight - Storm cluster	Microsoft.HDInsight/clusters	storm
HDInsight - ML Services cluster	Microsoft.HDInsight/clusters	mls
IoT hub	Microsoft.Devices/IotHubs	iot
Provisioning services	Microsoft.Devices/provisioningServices	provs
Provisioning services certificate	Microsoft.Devices/provisioningServices/certificates	pcert

Resource	Resource provider namespace	Abbreviation
Power BI Embedded	Microsoft.PowerBIDedicated/capacities	pbi
Time Series Insights environment	Microsoft.TimeSeriesInsights/environments	tsi

Compute and web

[\[+\]](#) Expand table

Resource	Resource provider namespace	Abbreviation
App Service environment	Microsoft.Web/hostingEnvironments	ase
App Service plan	Microsoft.Web/serverFarms	asp
Azure Load Testing instance	Microsoft.LoadTestService/loadTests	lt
Availability set	Microsoft.Compute/availabilitySets	avail
Azure Arc enabled server	Microsoft.HybridCompute/machines	arcs
Azure Arc enabled Kubernetes cluster	Microsoft.Kubernetes/connectedClusters	arck
Batch accounts	Microsoft.Batch/batchAccounts	ba
Cloud service	Microsoft.Compute/cloudServices	cld
Communication Services	Microsoft.Communication/communicationServices	acs
Disk encryption set	Microsoft.Compute/diskEncryptionSets	des
Function app	Microsoft.Web/sites	func
Gallery	Microsoft.Compute/galleries	gal
Hosting environment	Microsoft.Web/hostingEnvironments	host
Image template	Microsoft.VirtualMachineImages/imageTemplates	it
Managed disk (OS)	Microsoft.Compute/disks	osdisk

Resource	Resource provider namespace	Abbreviation
Managed disk (data)	Microsoft.Compute/disks	disk
Notification Hubs	Microsoft.NotificationHubs/namespaces/notificationHubs	ntf
Notification Hubs namespace	Microsoft.NotificationHubs/namespaces	ntfns
Proximity placement group	Microsoft.Compute/proximityPlacementGroups	ppg
Restore point collection	Microsoft.Compute/restorePointCollections	rpc
Snapshot	Microsoft.Compute/snapshots	snap
Static web app	Microsoft.Web/staticSites	stapp
Virtual machine	Microsoft.Compute/virtualMachines	vm
Virtual machine scale set	Microsoft.Compute/virtualMachineScaleSets	vmss
Virtual machine maintenance configuration	Microsoft.Maintenance/maintenanceConfigurations	mc
VM storage account	Microsoft.Storage/storageAccounts	stvm
Web app	Microsoft.Web/sites	app

Containers

[\[+\] Expand table](#)

Resource	Resource provider namespace	Abbreviation
AKS cluster	Microsoft.ContainerService/managedClusters	aks
AKS system node pool (mode: System)	Microsoft.ContainerService/managedClusters/agentPools	npsystem
AKS user node pool (mode: User)	Microsoft.ContainerService/managedClusters/agentPools	np
Container apps	Microsoft.App/containerApps	ca
Container apps	Microsoft.App/managedEnvironments	cae

Resource	Resource provider namespace	Abbreviation
environment		
Container registry	Microsoft.ContainerRegistry/registries	cr
Container instance	Microsoft.ContainerInstance/containerGroups	ci
Service Fabric cluster	Microsoft.ServiceFabric/clusters	sf
Service Fabric managed cluster	Microsoft.ServiceFabric/managedClusters	sfmc

Databases

[Expand table](#)

Resource	Resource provider namespace	Abbreviation
Azure Cosmos DB database	Microsoft.DocumentDB/databaseAccounts/sqlDatabases	cosmos
Azure Cosmos DB for Apache Cassandra account	Microsoft.DocumentDB/databaseAccounts	coscas
Azure Cosmos DB for MongoDB account	Microsoft.DocumentDB/databaseAccounts	cosmon
Azure Cosmos DB for NoSQL account	Microsoft.DocumentDb/databaseAccounts	cosno
Azure Cosmos DB for Table account	Microsoft.DocumentDb/databaseAccounts	costab
Azure Cosmos DB for Apache Gremlin account	Microsoft.DocumentDb/databaseAccounts	cosgrm
Azure Cosmos DB PostgreSQL cluster	Microsoft.DBforPostgreSQL/serverGroupsV2	cospos
Azure Cache for Redis instance	Microsoft.Cache/Redis	redis
Azure SQL Database server	Microsoft.Sql/servers	sql

Resource	Resource provider namespace	Abbreviation
Azure SQL database	Microsoft.Sql/servers/databases	sqlldb
Azure SQL Elastic Job agent	Microsoft.Sql/servers/jobAgents	sqlja
Azure SQL Elastic Pool	Microsoft.Sql/servers/elasticpool	sqlep
MariaDB server	Microsoft.DBforMariaDB/servers	maria
MariaDB database	Microsoft.DBforMariaDB/servers/databases	mariadb
MySQL database	Microsoft.DBforMySQL/servers	mysql
PostgreSQL database	Microsoft.DBforPostgreSQL/servers	psql
SQL Server Stretch Database	Microsoft.Sql/servers/databases	sqlstrdb
SQL Managed Instance	Microsoft.Sql/managedInstances	sqlmi

Developer tools

[\[+\] Expand table](#)

Resource	Resource provider namespace	Abbreviation
App Configuration store	Microsoft.AppConfiguration/configurationStores	appcs
Maps account	Microsoft.Maps/accounts	map
SignalR	Microsoft.SignalRService/SignalR	sigr
WebPubSub	Microsoft.SignalRService/webPubSub	wps

DevOps

[\[+\] Expand table](#)

Resource	Resource provider namespace	Abbreviation
Azure Managed Grafana	Microsoft.Dashboard/grafana	amg

Integration

[Expand table](#)

Resource	Resource provider namespace	Abbreviation
API management service instance	Microsoft.ApiManagement/service	apim
Integration account	Microsoft.Logic/integrationAccounts	ia
Logic app	Microsoft.Logic/workflows	logic
Service Bus namespace	Microsoft.ServiceBus/namespaces	s sns
Service Bus queue	Microsoft.ServiceBus/namespaces/queues	sbq
Service Bus topic	Microsoft.ServiceBus/namespaces/topics	sbt
Service Bus topic subscription	Microsoft.ServiceBus/namespaces/topics/subscriptions	sbts

Management and governance

[Expand table](#)

Resource	Resource provider namespace	Abbreviation
Automation account	Microsoft.Automation/automationAccounts	aa
Azure Policy definition	Microsoft.Authorization/policyDefinitions	<descriptive>
Application Insights	Microsoft.Insights/components	appi
Azure Monitor action group	Microsoft.Insights/actionGroups	ag
Azure Monitor data collection rule	Microsoft.Insights/dataCollectionRules	dcr
Azure Monitor alert processing rule	Microsoft.AlertsManagement/actionRules	apr
Blueprint (planned for deprecation)	Microsoft.Blueprint/blueprints	bp
Blueprint assignment (planned for deprecation)	Microsoft.Blueprint/blueprints/artifacts	bpa
Data collection endpoint	Microsoft.Insights/dataCollectionEndpoints	dce
Log Analytics workspace	Microsoft.OperationalInsights/worksheets	log

Resource	Resource provider namespace	Abbreviation
Log Analytics query packs	Microsoft.OperationalInsights/querypacks	pack
Management group	Microsoft.Management/managementGroups	mg
Microsoft Purview instance	Microsoft.Purview/accounts	pview
Resource group	Microsoft.Resources/resourceGroups	rg
Template specs name	Microsoft.Resources/templateSpecs	ts

Migration

[\[+\] Expand table](#)

Resource	Resource provider namespace	Abbreviation
Azure Migrate project	Microsoft.Migrate/assessmentProjects	migr
Database Migration Service instance	Microsoft.DataMigration/services	dms
Recovery Services vault	Microsoft.RecoveryServices/vaults	rsv

Networking

[\[+\] Expand table](#)

Resource	Resource provider namespace	Abbreviation
Application gateway	Microsoft.Network/applicationGateways	agw
Application security group (ASG)	Microsoft.Network/applicationSecurityGroups	asg
CDN profile	Microsoft.Cdn/profiles	cdnp
CDN endpoint	Microsoft.Cdn/profiles/endpoints	cdne
Connections	Microsoft.Network/connections	con
DNS	Microsoft.Network/dnsZones	<DNS domain name>

Resource	Resource provider namespace	Abbreviation
DNS forwarding ruleset	<code>Microsoft.Network/dnsForwardingRulesets</code>	<code>dnsfrs</code>
DNS private resolver	<code>Microsoft.Network/dnsResolvers</code>	<code>dnspr</code>
DNS private resolver inbound endpoint	<code>Microsoft.Network/dnsResolvers/inboundEndpoints</code>	<code>in</code>
DNS private resolver outbound endpoint	<code>Microsoft.Network/dnsResolvers/outboundEndpoints</code>	<code>out</code>
DNS zone	<code>Microsoft.Network/privateDnsZones</code>	<code><DNS domain name></code>
Firewall	<code>Microsoft.Network/azureFirewalls</code>	<code>afw</code>
Firewall policy	<code>Microsoft.Network/firewallPolicies</code>	<code>afwp</code>
ExpressRoute circuit	<code>Microsoft.Network/expressRouteCircuits</code>	<code>erc</code>
ExpressRoute gateway	<code>Microsoft.Network/virtualNetworkGateways</code>	<code>ergw</code>
Front Door (Standard/Premium) profile	<code>Microsoft.Cdn/profiles</code>	<code>afd</code>
Front Door (Standard/Premium) endpoint	<code>Microsoft.Cdn/profiles/afdEndpoints</code>	<code>fde</code>
Front Door firewall policy	<code>Microsoft.Network/frontdoorWebApplicationFirewallPolicies</code>	<code>fdfp</code>
Front Door (classic)	<code>Microsoft.Network/frontDoors</code>	<code>afd</code>
IP group	<code>Microsoft.Network/ipGroups</code>	<code>ipg</code>
Load balancer (internal)	<code>Microsoft.Network/loadBalancers</code>	<code>lbi</code>
Load balancer (external)	<code>Microsoft.Network/loadBalancers</code>	<code>lbe</code>
Load balancer rule	<code>Microsoft.Network/loadBalancers/inboundNatRules</code>	<code>rule</code>

Resource	Resource provider namespace	Abbreviation
Local network gateway	Microsoft.Network/localNetworkGateways	lgw
NAT gateway	Microsoft.Network/natGateways	ng
Network interface (NIC)	Microsoft.Network/networkInterfaces	nic
Network security group (NSG)	Microsoft.Network/networkSecurityGroups	nsg
Network security group (NSG) security rules	Microsoft.Network/networkSecurityGroups/securityRules	nsgsr
Network Watcher	Microsoft.Network/networkWatchers	nw
Private Link	Microsoft.Network/privateLinkServices	pl
Private endpoint	Microsoft.Network/privateEndpoints	pep
Public IP address	Microsoft.Network/publicIPAddresses	pip
Public IP address prefix	Microsoft.Network/publicIPPrefixes	ippre
Route filter	Microsoft.Network/routeFilters	rf
Route server	Microsoft.Network/virtualHubs	rtserv
Route table	Microsoft.Network/routeTables	rt
Service endpoint policy	Microsoft.serviceEndPointPolicies	se
Traffic Manager profile	Microsoft.Network/trafficManagerProfiles	traf
User defined route (UDR)	Microsoft.Network/routeTables/routes	udr
Virtual network	Microsoft.Network/virtualNetworks	vnet
Virtual network gateway	Microsoft.Network/virtualNetworkGateways	vgw
Virtual network manager	Microsoft.Network/networkManagers	vnm
Virtual network	Microsoft.Network/virtualNetworks/virtualNetworkPeerings	peer

Resource	Resource provider namespace	Abbreviation
peering		
Virtual network subnet	Microsoft.Network/virtualNetworks/subnets	snet
Virtual WAN	Microsoft.Network/virtualWans	vwan
Virtual WAN Hub	Microsoft.Network/virtualHubs	vhub

Security

[\[+\] Expand table](#)

Resource	Resource provider namespace	Abbreviation
Azure Bastion	Microsoft.Network/bastionHosts	bas
Key vault	Microsoft.KeyVault/vaults	kv
Key Vault Managed HSM	Microsoft.KeyVault/managedHSMs	kvmhsm
Managed identity	Microsoft.ManagedIdentity/userAssignedIdentities	id
SSH key	Microsoft.Compute/sshPublicKeys	sshkey
VPN Gateway	Microsoft.Network/vpnGateways	vpng
VPN connection	Microsoft.Network/vpnGateways/vpnConnections	vcn
VPN site	Microsoft.Network/vpnGateways/vpnSites	vst
Web Application Firewall (WAF) policy	Microsoft.Network/firewallPolicies	waf
Web Application Firewall (WAF) policy rule group	Microsoft.Network/firewallPolicies/ruleGroups	wafrg

Storage

[\[+\] Expand table](#)

Resource	Resource provider namespace	Abbreviation
Azure StorSimple	Microsoft.StorSimple/managers	ssimp

Resource	Resource provider namespace	Abbreviation
Backup Vault name	Microsoft.DataProtection/backupVaults	bvault
Backup Vault policy	Microsoft.DataProtection/backupVaults/backupPolicies	bkpol
File share	Microsoft.Storage/storageAccounts/fileServices/shares	share
Storage account	Microsoft.Storage/storageAccounts	st
Storage Sync Service name	Microsoft.StorageSync/storageSyncServices	sss

Virtual desktop infrastructure

[\[+\] Expand table](#)

Resource	Resource provider namespace	Abbreviation
Azure Lab Services lab plan	Microsoft.LabServices/labPlans	lp
Virtual desktop host pool	Microsoft.DesktopVirtualization/hostPools	vdpool
Virtual desktop application group	Microsoft.DesktopVirtualization/applicationGroups	vdag
Virtual desktop workspace	Microsoft.DesktopVirtualization/worksspaces	vdws
Virtual desktop scaling plan	Microsoft.DesktopVirtualization/scalingPlans	vdscaling

Next step

Review recommendations for tagging your Azure resources and assets.

[Define your tagging strategy](#)

Feedback

Was this page helpful?

[Yes](#)

[No](#)

Define your tagging strategy

Article • 05/29/2024

When you apply metadata tags to your cloud resources, you can include information that the resource name doesn't include, like information about the asset. You can use that information to run sophisticated filtering and reporting on resources. Include context about the resource's associated workload or application, operational requirements, and ownership information. IT or business teams use this information to find resources or generate reports about resource usage and billing.

The required tags and optional tags that you apply to resources differs among organizations. The following list provides examples of common tags that capture important context and information about a resource. Use this list as a starting point to establish your own tagging conventions.

Minimum suggested tags

Use the following tags to help guide the implementation and processes of Cloud Adoption Framework for Azure methodologies. The methodologies have many best practices that demonstrate cloud operations automation and governance based on the following tags.

[] Expand table

Tag name	Description	Key value and example values
Workload name	Name of the workload that the resource supports.	<i>WorkloadName</i> <ul style="list-style-type: none">• ControlCharts
Data classification	Sensitivity of data that the resource hosts.	<i>DataClassification</i> <ul style="list-style-type: none">• Non-business• Public• General• Confidential• Highly confidential
Business criticality	Business impact of the resource or supported workload.	<i>Criticality</i> <ul style="list-style-type: none">• Low• Medium

Tag name	Description	Key value and example values
		<ul style="list-style-type: none"> • High • Business unit-critical • Mission-critical
Business unit	<p>Top-level division of your company that owns the subscription or workload that the resource belongs to.</p> <p>In smaller organizations, this tag might represent a single corporate or shared top-level organizational element.</p>	<i>BusinessUnit</i> <ul style="list-style-type: none"> • Finance • Marketing • Product XYZ • Corp • Shared
Operations commitment	<p>Level of operations support provided for the workload or resource.</p>	<i>OpsCommitment</i> <ul style="list-style-type: none"> • Baseline only • Enhanced baseline • Platform operations • Workload operations
Operations team	<p>Team accountable for day-to-day operations.</p>	<i>OpsTeam</i> <ul style="list-style-type: none"> • Central IT • Cloud operations • Control Charts team • MSP-{Managed Service Provider name}

Other common tagging examples

Use the following tags to increase visibility into the usage of Azure resources.

[\[+\] Expand table](#)

Tag name	Description	Key and example values
Application name	<p>Added granularity, if the workload is subdivided across multiple applications or services.</p>	<i>ApplicationName</i>

Tag name	Description	Key and example values
		<ul style="list-style-type: none"> • IssueTrackingSystem
Approver name	Person responsible for approving costs related to the resource.	<i>Approver</i> <ul style="list-style-type: none"> • chris@contoso.com
Budget required/approved	Money approved for the application, service, or workload.	<i>BudgetAmount</i> <ul style="list-style-type: none"> • \$200,000
Cost center	Accounting cost center associated with the resource.	<i>CostCenter</i> <ul style="list-style-type: none"> • 55332
Disaster recovery	Business criticality of the application, workload, or service.	<i>DR</i> <ul style="list-style-type: none"> • Mission-critical • Critical • Essential
End date of the project	Date when the application, workload, or service is scheduled for retirement.	<i>EndDate</i> <ul style="list-style-type: none"> • 2023-10-15
Environment	Deployment environment of the application, workload, or service.	<i>Env</i> <ul style="list-style-type: none"> • Prod • Dev • QA • Stage • Test
Azure region	Region where you create the resource.	<i>AzureRegion</i> <ul style="list-style-type: none"> • West Europe • UK South • East US • Japan East • Qatar Central
Owner name	Owner of the application, workload, or service.	<i>Owner</i> <ul style="list-style-type: none"> • jane@contoso.com
Requester name	User who requested the creation of the application.	<i>Requester</i>

Tag name	Description	Key and example values
		<ul style="list-style-type: none"> • <code>john@contoso.com</code>
Service class	Service-level agreement level of the application, workload, or service.	<code>ServiceClass</code> <ul style="list-style-type: none"> • <code>Dev</code> • <code>Bronze</code> • <code>Silver</code> • <code>Gold</code>
Start date of the project	Date when the application, workload, or service was first deployed.	<code>StartDate</code> <ul style="list-style-type: none"> • <code>2020-10-15</code>

Take action

Review the [resource naming and tagging decision guide](#).

Next step

Learn how to move resource groups and assets between subscriptions in Azure.

[Move resource groups and assets between subscriptions](#)

Feedback

Was this page helpful?

Yes

No

Resource naming and tagging decision guide

Article • 05/29/2024

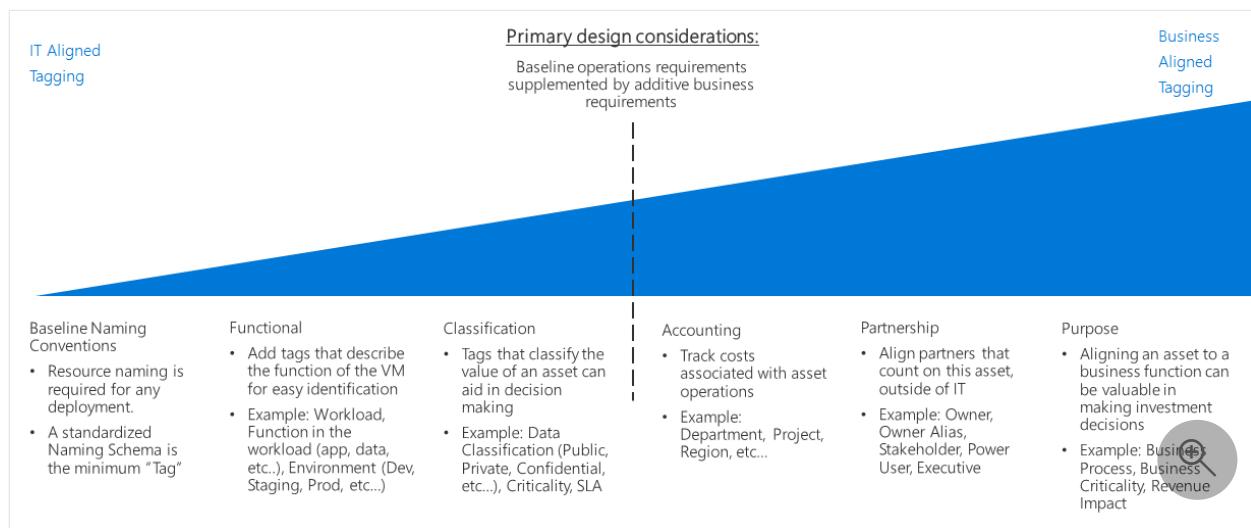
Organize your cloud-based resources so that your IT team can easily work with them. For complicated or complex deployments, use naming and tagging standards to organize your resources for:

- **Resource management:** Your IT teams need to quickly locate resources that are associated with specific workloads, regions, environments, ownership groups, or other important information. Organize resources to ensure that you properly assign organizational roles and access permissions.
- **Cost management and optimization:** Ensure that your IT team understands the resources and workloads that each team uses so that business groups know how much cloud resources consume. Cost-related tags support the following types of information:
 - [Cloud accounting models](#)
 - [Return on investment \(ROI\) calculations](#)
 - [Cost tracking](#)
 - [Budgets](#)
 - [Alerts](#)
 - [Recurring spend tracking and reporting](#)
 - [Post-implementation optimizations](#)
 - [Cost-optimization tactics](#)
- **Operations management:** Ensure that the operations management team has visibility into business commitments and service-level agreements (SLAs) for ongoing operations. Create tags for [mission criticality](#) to properly manage operations.
- **Security:** Classify data and determine the security impact to prepare for breaches or other security problems. Create tags for [data classification](#) to ensure that your operation is secure.
- **Governance and regulatory compliance:** Maintain consistency across resources to help identify divergence from policies. [Prescriptive guidance for resource tagging](#) demonstrates how one of the tagging patterns that are described in [Resource tagging patterns](#), later in this article, can help with deployment of governance practices. Similar patterns are available to evaluate regulatory compliance by using tags.

- **Automation:** Have a proper organizational scheme so you can use automation to create resources, monitor operations, and create DevOps processes. Automation also makes resources easier for IT to manage.
- **Workload optimization:** Use tagging to help resolve broad problems, identify patterns, and identify the assets that a single workload requires. Tag all assets that are associated with each workload so that you can deeply analyze your mission-critical workloads, which helps you make sound architectural decisions.

Tagging decision guide

You can have a simple or complex approach to tagging. Your approach can support IT teams that manage cloud workloads, or your approach can integrate information that's related to all aspects of the business.



The following table describes the considerations in the diagram.

[Expand table](#)

Consideration	Description
Primary design considerations	Baseline operations requirements, supplemented by additive business requirements.
Baseline naming conventions	Resource naming is required for deployment. A standardized naming schema is the minimum tag.
Functional	<p>Tags that describe the function of the virtual machine for easy identification.</p> <p>For example, a workload tag might describe the function in the workload, such as app or data. An environment tag might describe a function, such as development, staging, or production.</p>

Consideration	Description
Classification	Tags that classify the value of an asset can help you make decisions. For example, you can classify resources based on the data classification (public, private, or confidential), criticality, or SLAs.
Accounting	Tags that help track costs that are associated with asset operations. For example, use tags based on the department, project, or region.
Purpose	Tags that align an asset to a business function can be valuable in making investment decisions. For example, use tags based on the business process, business criticality, or revenue impact.

A tagging scheme that aligns with IT, such as tagging based on the workload, application, environment, or region, reduces the complexity of monitoring assets. With less complexity, you can simplify the process of making management decisions that are based on operational requirements.

Tagging schemes that align with business, like accounting, business ownership, or business criticality, might require a larger investment of time. You need to invest more time to create tagging standards that reflect business interests and maintain those standards in the future. This investment yields a tagging system that provides improved accounting for costs and value of IT assets to the overall business. Linking an asset's business value to its operational cost can change the view of IT as a cost center within your wider organization.

Baseline naming conventions

Use a standardized naming convention as a starting point to organize your cloud-hosted resources. When you have a properly structured naming system, you can quickly identify resources for both management and accounting purposes. You might have existing IT-aligned naming conventions in other parts of your organization. If so, consider whether your cloud naming conventions should align with them, or if you should establish separate cloud-based standards.

 Note

[Naming rules and restrictions](#) vary depending on the Azure resource. Your naming conventions must comply with these rules.

Resource tagging patterns

In addition to consistent naming conventions, cloud platforms also support the ability to tag resources, which provides more extensive organization.

Tags are metadata elements that are attached to resources and are valid across all regions under your tenant. Tags consist of pairs of key-value strings. The values that you include in these pairs are based on the requirements of your business. For more information, see [Minimum suggested tags](#). When you incorporate your comprehensive naming and tagging policy, apply a consistent set of global tags for overall governance.

When you plan for tagging, consider the following questions to determine the kind of information that your resource tags must support:

- Do your naming and tagging policies need to integrate with existing policies within your company?
- Will you implement a chargeback or showback accounting system? Do you need to associate resources with accounting information for departments, business groups, and teams in more detail than a simple subscription-level breakdown provides?
- Should tags represent details for a resource, such as regulatory compliance requirements? What about operational details such as uptime requirements, patching schedules, or security requirements?
- What tags are required for all resources based on centralized IT policy? What tags are optional? Are individual teams allowed to implement their own custom tagging schemes?

The following tagging patterns are examples of how you can use tagging to organize cloud assets. These patterns aren't meant to be exclusive, and you can use them in parallel. They provide multiple ways of organizing assets based on your company's needs.

[] [Expand table](#)

Tag type	Examples	Description
Functional	<code>app = catalogsearch1</code> <code>tier = web</code> <code>webserver = apache</code> <code>env = prod</code> <code>env = staging</code> <code>env = dev</code> <code>region = eastus</code> <code>region = uksouth</code>	Categorizes resources by their purposes within a workload, the environment and region they're deployed to, or other functionality and operational details

Tag type	Examples	Description
Classification	<code>confidentiality = private</code> <code>SLA = 24hours</code>	Classifies a resource by how it's used and the policies that apply to it
Accounting	<code>department = finance</code> <code>program = business-initiative</code> <code>region = northamerica</code>	Associates a resource with specific groups within an organization for billing purposes
Purpose	<code>businessprocess = support</code> <code>businessimpact = moderate</code> <code>revenueimpact = high</code>	Aligns resources to business functions to better support investment decisions

Multiregion resource tagging

You can use Azure tags across various Azure regions to logically organize resources. Azure tags aren't tied to a specific location, so you can use the same tagging strategy across all your resources regardless of their location.

In a multiregion environment, consider including region details in your tagging strategy if your naming convention doesn't already cover operational and management requirements.

You can also use tagging to aggregate and compare resources across regions and subscriptions. For example, you might require advanced reporting or resource filtering based on the Azure region where you deploy resources. If you can't align subscriptions to acquire these capabilities, you can use tagging instead.

If the region where you create a resource is a resource object property, you don't need to tag the resource.

Azure has built-in policies to enforce tagging requirements. You can also create custom policies for more specific tagging requirements. For more information, see [Assign policy definitions for tag compliance](#).

When you create an assignment for a policy, you can specify a resource selector, such as `resourceLocation`, to target and filter specific regions within a specified scope. For more information, see [Create a policy assignment](#) and [Resource selectors](#).

Next steps

- Resource tagging is just one of the core infrastructure components that requires architectural decisions in a process of cloud adoption. To learn about alternative patterns or models for making design decisions about other types of infrastructure, see the [architectural decision guides](#).
 - For recommended naming conventions for Azure resources, see [Develop your naming and tagging strategy for Azure resources](#).
 - For information about applying tags at both the resource group level and individual resource level, see [Use tags to organize your Azure resources and management hierarchy](#). Use this approach for flexibility in the granularity of accounting reports that are based on applied tags.
-

Feedback

Was this page helpful?

 Yes

 No

Move Azure resources to a new resource group or subscription

Article • 05/31/2024

This article shows you how to move Azure resources to either another Azure subscription or another resource group under the same subscription. You can use the Azure portal, Azure PowerShell, Azure CLI, or the REST API to move resources.

Both the source group and the target group are locked during the move operation. Write and delete operations are blocked on the resource groups until the move completes. This lock means you can't add, update, or delete resources in the resource groups. It doesn't mean the resources are frozen. For example, if you move an Azure SQL logical server, its databases and other dependent resources to a new resource group or subscription, applications that use the databases experience no downtime. They can still read and write to the databases. The lock can last for a maximum of four hours, but most moves complete in much less time.

If your move requires setting up new dependent resources, you'll experience an interruption in those services until they've been reconfigured.

Moving a resource only moves it to a new resource group or subscription. It doesn't change the location of the resource.

Changed resource ID

When you move a resource, you change its resource ID. The standard format for a resource ID is

`/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/{resourceProviderNamespace}/{resourceType}/{resourceName}`. When you move a resource to a new resource group or subscription, you change one or more values in that path.

If you use the resource ID anywhere, you'll need to change that value. For example, if you have a [custom dashboard](#) in the portal that references a resource ID, you'll need to update that value. Look for any scripts or templates that need to be updated for the new resource ID.

Checklist before moving resources

There are some important steps to do before moving a resource. By verifying these conditions, you can avoid errors.

1. The source and destination subscriptions must be active. If you have trouble enabling an account that has been disabled, [create an Azure support request](#). Select **Subscription Management** for the issue type.
2. The source and destination subscriptions must exist within the same [Microsoft Entra tenant](#). To check that both subscriptions have the same tenant ID, use Azure PowerShell or Azure CLI.

For Azure PowerShell, use:

Azure PowerShell

```
(Get-AzSubscription -SubscriptionName <your-source-subscription>).TenantId  
(Get-AzSubscription -SubscriptionName <your-destination-subscription>).TenantId
```

For Azure CLI, use:

Azure CLI

```
az account show --subscription <your-source-subscription> --query tenantId  
az account show --subscription <your-destination-subscription> --query tenantId
```

If the tenant IDs for the source and destination subscriptions aren't the same, use the following methods to reconcile the tenant IDs:

- [Transfer ownership of an Azure subscription to another account](#)
- [How to associate or add an Azure subscription to Microsoft Entra ID](#)

3. If you're attempting to move resources to or from a Cloud Solution Provider (CSP) partner, see [Transfer Azure subscriptions between subscribers and CSPs](#).
4. The resources you want to move must support the move operation. For a list of which resources support move, see [Move operation support for resources](#).
5. Some services have specific limitations or requirements when moving resources. If you're moving any of the following services, check that guidance before moving.
 - If you're using Azure Stack Hub, you can't move resources between groups.
 - [App Services move guidance](#)

- Azure DevOps Services move guidance
- [Classic deployment model move guidance](#) - Classic Compute, Classic Storage, Classic Virtual Networks, and Cloud Services
- [Cloud Services \(extended support\) move guidance](#)
- [Networking move guidance](#)
- [Recovery Services move guidance](#)
- [Virtual Machines move guidance](#)
- To move an Azure subscription to a new management group, see [Move subscriptions](#).

6. The destination subscription must be registered for the resource provider of the resource being moved. If not, you receive an error stating that the **subscription is not registered for a resource type**. You might see this error when moving a resource to a new subscription, but that subscription has never been used with that resource type.

For PowerShell, use the following commands to get the registration status:

Azure PowerShell

```
Set-AzContext -Subscription <destination-subscription-name-or-id>
Get-AzResourceProvider -ListAvailable | Select-Object
ProviderNamespace, RegistrationState
```

To register a resource provider, use:

Azure PowerShell

```
Register-AzResourceProvider -ProviderNamespace Microsoft.Batch
```

For Azure CLI, use the following commands to get the registration status:

Azure CLI

```
az account set -s <destination-subscription-name-or-id>
az provider list --query "[].{Provider:namespace,
Status:registrationState}" --out table
```

To register a resource provider, use:

Azure CLI

```
az provider register --namespace Microsoft.Batch
```

7. Before moving the resources, check the subscription quotas for the subscription you're moving the resources to. If moving the resources means the subscription will exceed its limits, you need to review whether you can request an increase in the quota. For a list of limits and how to request an increase, see [Azure subscription and service limits, quotas, and constraints](#).

8. The account moving the resources must have at least the following permissions:

- `Microsoft.Resources/subscriptions/resourceGroups/moveResources/action` on the source resource group.
- `Microsoft.Resources/subscriptions/resourceGroups/write` on the destination resource group.

9. If you move a resource that has an Azure role assigned directly to the resource (or a child resource), the role assignment isn't moved and becomes orphaned. After the move, you must re-create the role assignment. Eventually, the orphaned role assignment is automatically removed, but we recommend removing the role assignment before the move.

For information about how to manage role assignments, see [List Azure role assignments](#) and [Assign Azure roles](#).

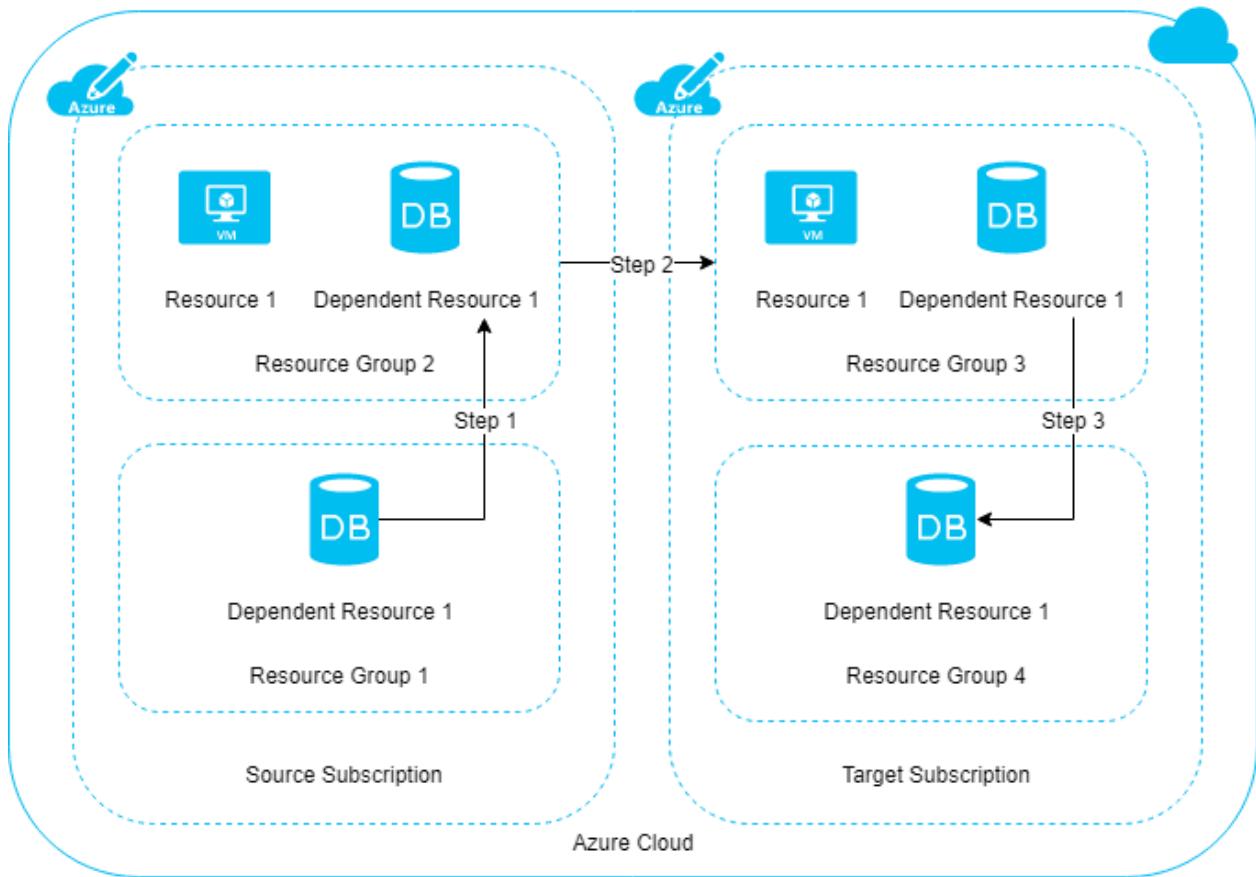
10. **For a move across subscriptions, the resource and its dependent resources must be located in the same resource group and they must be moved together.** For example, a VM with managed disks would require the VM and the managed disks to be moved together, along with other dependent resources.

If you're moving a resource to a new subscription, check to see whether the resource has any dependent resources, and whether they're located in the same resource group. If the resources aren't in the same resource group, check to see whether the resources can be combined into the same resource group. If so, bring all these resources into the same resource group by using a move operation across resource groups.

For more information, see [Scenario for move across subscriptions](#).

Scenario for move across subscriptions

Moving resources from one subscription to another is a three-step process:



For illustration purposes, we have only one dependent resource.

- Step 1: If dependent resources are distributed across different resource groups, first move them into one resource group.
- Step 2: Move the resource and dependent resources together from the source subscription to the target subscription.
- Step 3: Optionally, redistribute the dependent resources to different resource groups within the target subscription.

Use the portal

To move resources, select the resource group that contains those resources.

Select the resources you want to move. To move all of the resources, select the checkbox at the top of list. Or, select resources individually.

The screenshot shows the Microsoft Azure Resource Groups blade for a resource group named "sourceGroup". The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Events, Deployments, Security, Policies, Properties, Locks, Cost analysis, and Cost alerts (preview). The main area displays the "Essentials" section with information about the subscription (change) and tags. Below this is a table listing resources, with the "Name" column header highlighted by a red box. The listed resources are:

Name
exampleVM1
exampleVM1-ip
exampleVM1-nsg
examplevm1920
exampleVM1_OsDisk_1_38427735e90a4270a5888d64e37065de
sourceGroup-vnet

Select the **Move** button.

The screenshot shows the Microsoft Azure Resource Groups blade for the same resource group. The top right corner features a "Move" button with a dropdown arrow. A context menu is open from this button, containing three options: "Move to another resource group", "Move to another subscription", and "Move to another region".

This button gives you three options:

- Move to a new resource group.
- Move to a new subscription.
- Move to a new region. To change regions, see [Move resources across regions \(from resource group\)](#).

Select whether you're moving the resources to a new resource group or a new subscription.

The source resource group is automatically set. Specify the destination resource group. If you're moving to a new subscription, also specify the subscription. Select **Next**.

Move resources

sourceGroup

1 Source + target **2** Resources to move **3** Review

To move a resource, select a source and a destination. The source and destination resource groups will both be locked during the move. [Learn more](#)

Source

Subscription Documentation Testing 1

Resource group sourceGroup

Target

Subscription Documentation Testing 1

Resource group * destinationGroup
[Create new](#)

[Previous](#)[Next](#)

The portal validates that the resources can be moved. Wait for validation to complete.

Move resources

sourceGroup

1 Source + target **2** Resources to move **3** Review

i Checking whether these resources can be moved. This might take a few minutes. **↻**

[+ Add resources](#)[X Remove from the move list](#)

Name	Type	Resource type	Validation status
exampleVM1	Virtual machine	microsoft.compute/virtualmachines	Pending validation
exampleVM1-ip	Public IP address	microsoft.network/publicipaddresses	Pending validation
exampleVM1-nsg	Network security group	microsoft.network/networksecuritygroups	Pending validation
examplevm1920	Network interface	microsoft.network/networkinterfaces	Pending validation
exampleVM1_OsDisk_1_38427735e90a427l	Disk	microsoft.compute/disks	Pending validation
sourceGroup-vnet	Virtual network	microsoft.network/virtualnetworks	Pending validation

When validation completes successfully, select **Next**.

Acknowledge that you need to update tools and scripts for these resources. To start moving the resources, select **Move**.

Move resources

sourceGroup

1 Source + target 2 Resources to move 3 Review

Selection summary

Source subscription	Documentation Testing 1
Source resource group	sourceGroup
Target subscription	Documentation Testing 1
Target resource group	destinationGroup
Number of resources to move	6

I understand that tools and scripts associated with moved resources will not work until I update them to use new resource IDs

Previous

Move

When the move has completed, you're notified of the result.

The screenshot shows the Azure Notifications center. At the top, there's a blue header bar with icons for search, refresh, notifications (which is highlighted with a red box), settings, help, and user profile. Below the header, the title "Notifications" is displayed. In the main area, there's a message: "More events in the activity log →" followed by a "Dismiss all" button. A single notification card is shown, indicating a successful move: "Moving resources complete" with a checkmark icon. The message details: "Successfully moved 6 resources from resource group 'sourceGroup' in subscription 'Documentation Testing 1' to resource group 'destinationGroup' in subscription 'Documentation Testing 1'". Below the message are "Feedback" and "Related events" buttons, and the timestamp "12 minutes ago".

Use Azure PowerShell

Validate

To test your move scenario without actually moving the resources, use the [Invoke-AzResourceAction](#) command. Use this command only when you need to predetermine the results.

```
Azure PowerShell

$sourceName = "sourceRG"
$destinationName = "destinationRG"
$resourcesToMove = @("app1", "app2")

$sourceResourceGroup = Get-AzResourceGroup -Name $sourceName
$destinationResourceGroup = Get-AzResourceGroup -Name $destinationName

$resources = Get-AzResource -ResourceGroupName $sourceName | Where-Object {
    $_.Name -in $resourcesToMove }

Invoke-AzResourceAction -Action validateMoveResources ` 
    -ResourceId $sourceResourceGroup.ResourceId ` 
    -Parameters @{
        resources = $resources.ResourceId; # Wrap in an @() array if
        providing a single resource ID string.
        targetResourceGroup = $destinationResourceGroup.ResourceId
    }
```

If validation passes, you see no output.

If validation fails, you see an error message describing why the resources can't be moved.

Move

To move existing resources to another resource group or subscription, use the [Move-AzResource](#) command. The following example shows how to move several resources to a new resource group.

```
Azure PowerShell

$sourceName = "sourceRG"
$destinationName = "destinationRG"
$resourcesToMove = @("app1", "app2")

$resources = Get-AzResource -ResourceGroupName $sourceName | Where-Object {
    $_.Name -in $resourcesToMove }

Move-AzResource -DestinationResourceGroupName $destinationName -ResourceId
$resources.ResourceId
```

To move to a new subscription, include a value for the `DestinationSubscriptionId` parameter.

Use Azure CLI

Validate

To test your move scenario without actually moving the resources, use the [az resource invoke-action](#) command. Use this command only when you need to predetermine the results. To run this operation, you need the:

- Resource ID of the source resource group
- Resource ID of the target resource group
- Resource ID of each resource to move

In the request body, use `\"` to escape double quotes.

```
Azure CLI

az resource invoke-action --action validateMoveResources \
--ids "/subscriptions/{subscription-id}/resourceGroups/{source-rg}" \
--request-body "{ \"resources\": [\"/subscriptions/{subscription-
id}/resourceGroups/{source-rg}/providers/{resource-provider}/{resource-
type}/{resource-name}\", \"/subscriptions/{subscription-
id}/resourceGroups/{source-rg}/providers/{resource-provider}/{resource-
type}/{resource-name}\", \"/subscriptions/{subscription-
id}/resourceGroups/{source-rg}/providers/{resource-provider}/{resource-
type}/{resource-name}\"] }, \"targetResourceGroup\": \"/subscriptions/{subscription-
id}/resourceGroups/{destination-rg}\" }"
```

If validation passes, you see:

```
Azure CLI

{} Finished ..
```

If validation fails, you see an error message describing why the resources can't be moved.

Move

To move existing resources to another resource group or subscription, use the [az resource move](#) command. In the `--ids` parameter, provide a space-separated list of the

resource IDs to move.

The following example shows how to move several resources to a new resource group. It works when using Azure CLI in a **Bash** terminal.

Azure CLI

```
webapp=$(az resource show -g OldRG -n ExampleSite --resource-type "Microsoft.Web/sites" --query id --output tsv)
plan=$(az resource show -g OldRG -n ExamplePlan --resource-type "Microsoft.Web/serverfarms" --query id --output tsv)
az resource move --destination-group newgroup --ids $webapp $plan
```

The next example shows how to run the same commands in a **PowerShell** console.

Azure CLI

```
$webapp=$(az resource show -g OldRG -n ExampleSite --resource-type "Microsoft.Web/sites" --query id --output tsv)
$plan=$(az resource show -g OldRG -n ExamplePlan --resource-type "Microsoft.Web/serverfarms" --query id --output tsv)
az resource move --destination-group newgroup --ids $webapp $plan
```

To move to a new subscription, provide the `--destination-subscription-id` parameter.

Use Python

Validate

To test your move scenario without actually moving the resources, use the [ResourceManagementClient.resources.begin_validate_move_resources](#) method. Use this method only when you need to predetermine the results.

Python

```
import os
from azure.identity import AzureCliCredential
from azure.mgmt.resource import ResourceManagementClient

credential = AzureCliCredential()
subscription_id = os.environ["AZURE_SUBSCRIPTION_ID"]

resource_client = ResourceManagementClient(credential, subscription_id)

source_name = "sourceRG"
destination_name = "destinationRG"
```

```

resources_to_move = ["app1", "app2"]

destination_resource_group =
resource_client.resource_groups.get(destination_name)

resources = [
    resource for resource in
resource_client.resources.list_by_resource_group(source_name)
    if resource.name in resources_to_move
]

resource_ids = [resource.id for resource in resources]

validate_move_resources_result =
resource_client.resources.begin_validate_move_resources(
    source_name,
{
    "resources": resource_ids,
    "target_resource_group": destination_resource_group.id
}
).result()

print("Validate move resources result:
{}".format(validate_move_resources_result))

```

If validation passes, you see no output.

If validation fails, you see an error message describing why the resources can't be moved.

Move

To move existing resources to another resource group or subscription, use the [ResourceManagementClient.resources.begin_move_resources](#) method. The following example shows how to move several resources to a new resource group.

Python

```

import os
from azure.identity import AzureCliCredential
from azure.mgmt.resource import ResourceManagementClient

credential = AzureCliCredential()
subscription_id = os.environ["AZURE_SUBSCRIPTION_ID"]

resource_client = ResourceManagementClient(credential, subscription_id)

source_name = "sourceRG"
destination_name = "destinationRG"
resources_to_move = ["app1", "app2"]

```

```

destination_resource_group =
resource_client.resource_groups.get(destination_name)

resources = [
    resource for resource in
resource_client.resources.list_by_resource_group(source_name)
    if resource.name in resources_to_move
]

resource_ids = [resource.id for resource in resources]

resource_client.resources.begin_move_resources(
    source_name,
    {
        "resources": resource_ids,
        "target_resource_group": destination_resource_group.id
    }
)

```

Use REST API

Validate

The [validate move operation](#) lets you test your move scenario without actually moving the resources. Use this operation to check if the move will succeed. Validation is automatically called when you send a move request. Use this operation only when you need to predetermine the results. To run this operation, you need the:

- Name of the source resource group
- Resource ID of the target resource group
- Resource ID of each resource to move
- The [access token](#) for your account

Send the following request:

HTTP

```

POST https://management.azure.com/subscriptions/<subscription-
id>/resourceGroups/<source-group>/validateMoveResources?api-version=2019-05-
10
Authorization: Bearer <access-token>
Content-type: application/json

```

With a request body:

JSON

```
{  
  "resources": ["<resource-id-1>", "<resource-id-2>"],  
  "targetResourceGroup": "/subscriptions/<subscription-  
id>/resourceGroups/<target-group>"  
}
```

If the request is formatted correctly, the operation returns:

HTTP

```
Response Code: 202  
cache-control: no-cache  
pragma: no-cache  
expires: -1  
location: https://management.azure.com/subscriptions/<subscription-  
id>/operationresults/<operation-id>?api-version=2018-02-01  
retry-after: 15  
...
```

The 202 status code indicates the validation request was accepted, but it hasn't yet determined if the move operation will succeed. The `location` value contains a URL that you use to check the status of the long-running operation.

To check the status, send the following request:

HTTP

```
GET <location-url>  
Authorization: Bearer <access-token>
```

While the operation is still running, you continue to receive the 202 status code. Wait the number of seconds indicated in the `retry-after` value before trying again. If the move operation validates successfully, you receive the 204 status code. If the move validation fails, you receive an error message, such as:

JSON

```
{"error":{"code":"ResourceMoveProviderValidationFailed","message":  
<message>}...}}
```

Move

To move existing resources to another resource group or subscription, use the [Move resources](#) operation.

HTTP

```
POST https://management.azure.com/subscriptions/{source-subscription-id}/resourcegroups/{source-resource-group-name}/moveResources?api-version={api-version}
```

In the request body, you specify the target resource group and the resources to move.

JSON

```
{  
  "resources": ["<resource-id-1>", "<resource-id-2>"],  
  "targetResourceGroup": "/subscriptions/<subscription-id>/resourceGroups/<target-group>"  
}
```

Frequently asked questions

Question: My resource move operation, which usually takes a few minutes, has been running for almost an hour. Is there something wrong?

Moving a resource is a complex operation that has different phases. It can involve more than just the resource provider of the resource you're trying to move. Because of the dependencies between resource providers, Azure Resource Manager allows 4 hours for the operation to complete. This time period gives resource providers a chance to recover from transient issues. If your move request is within the four-hour period, the operation keeps trying to complete and may still succeed. The source and destination resource groups are locked during this time to avoid consistency issues.

Question: Why is my resource group locked for four hours during resource move?

A move request is allowed a maximum of four hours to complete. To prevent modifications on the resources being moved, both the source and destination resource groups are locked during the resource move.

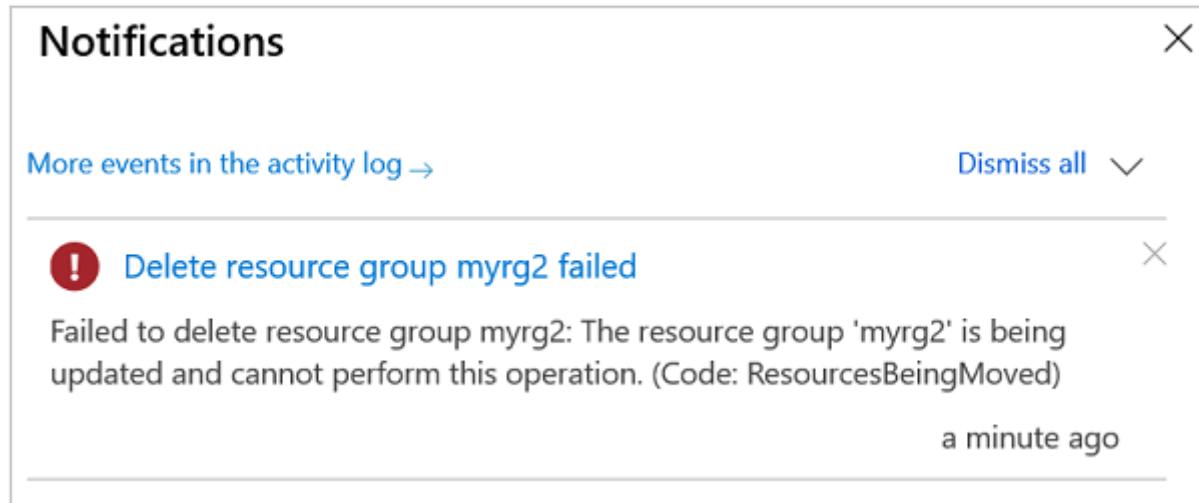
There are two phases in a move request. In the first phase, the resource is moved. In the second phase, notifications are sent to other resource providers that are dependent on the resource being moved. A resource group can be locked for the entire four hours when a resource provider fails either phase. During the allowed time, Resource Manager retries the failed step.

If a resource can't be moved within four hours, Resource Manager unlocks both resource groups. Resources that were successfully moved are in the destination resource group. Resources that failed to move are left the source resource group.

Question: What are the implications of the source and destination resource groups being locked during the resource move?

The lock prevents you from deleting either resource group, creating a new resource in either resource group, or deleting any of the resources involved in the move.

The following image shows an error message from the Azure portal when a user tries to delete a resource group that is part of an ongoing move.



Question: What does the error code "MissingMoveDependentResources" mean?

When you move a resource, its dependent resources must either exist in the destination resource group or subscription, or be included in the move request. You get the MissingMoveDependentResources error code when a dependent resource doesn't meet this requirement. The error message has details about the dependent resource that needs to be included in the move request.

For example, moving a virtual machine could require moving seven resource types with three different resource providers. Those resource providers and types are:

- Microsoft.Compute
 - virtualMachines
 - disks
- Microsoft.Network
 - networkInterfaces
 - publicIPAddresses
 - networkSecurityGroups
 - virtualNetworks
- Microsoft.Storage
 - storageAccounts

Another common example involves moving a virtual network. You may have to move several other resources associated with that virtual network. The move request could require moving public IP addresses, route tables, virtual network gateways, network security groups, and others. In general, a virtual network gateway must always be in the same resource group as its virtual network, they can't be moved separately.

Question: What does the error code "RequestDisallowedByPolicy" mean?

Resource Manager validates your move request before attempting the move. This validation includes checking policies defined on the resources involved in the move. For example, if you're attempting to move a key vault but your organization has a policy to deny the creation of a key vault in the target resource group, validation fails and the move is blocked. The returned error code is **RequestDisallowedByPolicy**.

For more information about policies, see [What is Azure Policy?](#).

Question: Why can't I move some resources in Azure?

Currently, not all resources in Azure support move. For a list of resources that support move, see [Move operation support for resources](#).

Question: How many resources can I move in a single operation?

When possible, break large moves into separate move operations. Resource Manager immediately returns an error when there are more than 800 resources in a single operation. However, moving less than 800 resources may also fail by timing out.

Question: What is the meaning of the error that a resource isn't in succeeded state?

When you get an error message that indicates a resource can't be moved because it isn't in a succeeded state, it may actually be a dependent resource that is blocking the move. Typically, the error code is

MoveCannotProceedWithResourcesNotInSucceededState.

If the source or target resource group contains a virtual network, the states of all dependent resources for the virtual network are checked during the move. The check includes those resources directly and indirectly dependent on the virtual network. If any of those resources are in a failed state, the move is blocked. For example, if a virtual machine that uses the virtual network has failed, the move is blocked. The move is blocked even when the virtual machine isn't one of the resources being moved and isn't in one of the resource groups for the move.

When you receive this error, you have two options. Either move your resources to a resource group that doesn't have a virtual network, or [contact support](#).

Question: Can I move a resource group to a different subscription?

No, you can't move a resource group to a new subscription. But, you can move all of the resources in the resource group to a resource group in another subscription. Settings such as tags, role assignments, and policies aren't automatically transferred from the original resource group to the destination resource group. You need to reapply these settings to the new resource group. For more information, see [Move resources to new resource group or subscription](#).

Next steps

For a list of which resources support move, see [Move operation support for resources](#).

ⓘ **Note:** The author created this article with assistance from AI. [Learn more](#)

Create your initial Azure subscriptions

Article • 06/06/2023

Begin your Azure adoption process by creating a set of subscriptions based on your organization's initial requirements.

ⓘ Note

Use the Azure landing zone guidance for **resource organization** as a first step towards planning subscriptions within your Azure environment to ensure you consider environment scaling.

Create subscriptions

Create two Azure subscriptions:

- A subscription that contains your production workloads.
- A subscription that serves as your non-production environment, using an [Azure Dev/Test offer ↗](#) for lower pricing.



Figure 1: An initial subscription model with keys next to boxes labeled "production" and "nonproduction".

A two-subscription approach offers many benefits:

- The use of separate subscriptions for production and non-production environments creates a boundary that makes resource management simpler and safer.
- Azure Dev/Test subscription offerings are available for non-production workloads. These offerings provide discounted rates on Azure services and software licensing.
- Production and non-production environments often have different sets of Azure policies. Placing each environment in its own subscription makes it simple for you to apply different policies to them at the subscription level.

- You can place certain types of Azure resources in a non-production subscription for testing purposes. You can enable resource providers for these test resources in your non-production subscription without ever exposing them to your production environment.
- You can use Azure dev/test subscriptions as isolated sandbox environments. These sandboxes allow administrators and developers to rapidly create and tear down entire sets of Azure resources and help with data protection and security concerns.
- Acceptable cost thresholds often vary between production and non-production environments.

Sandbox subscriptions

If you know your organization's cloud adoption strategy requires innovation, consider creating one or more [sandbox subscriptions](#). In sandbox subscriptions, you can experiment with Azure capabilities and apply security policies to keep test subscriptions isolated from your production and non-production environments. Use an Azure Dev/Test offer to create these subscriptions.



- *Figure 2: A subscription model with sandbox subscriptions.*

Shared services subscriptions

If your organization plans to host **more than 1,000 VMs or compute instances in the cloud within 24 months**, you should create another Azure subscription to host shared services. This strategy helps prepare you to support your end-state enterprise architecture.

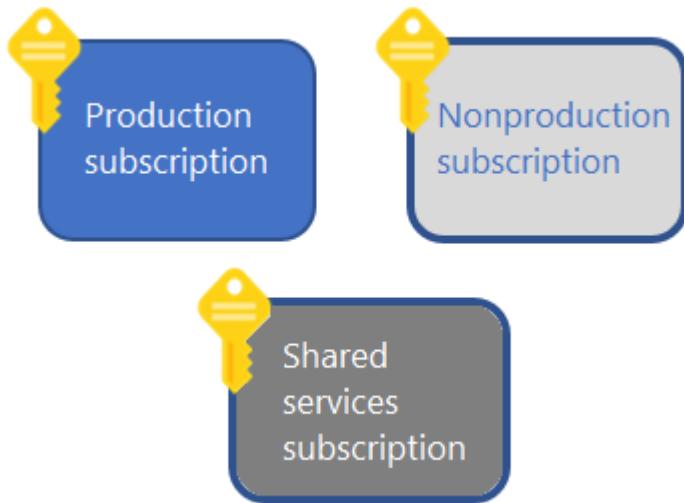


Figure 3: A subscription model with shared services.

Next steps

[Scale your Azure environment with additional subscriptions](#)

Create additional subscriptions to scale your Azure environment

Article • 01/10/2023

Organizations often use multiple Azure subscriptions to avoid per-subscription resource limits and to better manage and govern their Azure resources. It's important to define a strategy for scaling your subscriptions.

ⓘ Note

We recommend that organizations consider the Azure landing zone guidance for **resource organization** as a first step to planning subscriptions within an Azure environment to ensure the broader context of an environment intended to scale is considered

Review fundamental concepts

As you expand your Azure environment beyond your [initial subscriptions](#), it's important to understand Azure concepts such as accounts, tenants, directories, and subscriptions. For more information, see [Azure fundamental concepts](#).

Other considerations might necessitate additional subscriptions. Keep the following in mind as you expand your cloud estate.

Technical considerations

Subscription limits: Subscriptions have defined limits for some resource types. For example, the number of virtual networks in a subscription is limited. When a subscription approaches these limits, you'll need to create another subscription and put additional resources there. For more information, see [Azure subscription and service limits](#).

Classic model resources: If you've been using Azure for a long time, you may have resources that were created using the classic deployment model. Azure policies, Azure role-based access control, resource grouping, and tags cannot be applied to classic model resources. You should move these resources into subscriptions that contain only classic model resources.

Costs: There might be some additional costs for data ingress and egress between subscriptions.

Business priorities

Your business priorities might lead you to create additional subscriptions. These priorities include:

- Innovation
- Migration
- Cost
- Operations
- Security
- Governance

For other considerations about scaling your subscriptions, review the [subscription organization and governance recommendations](#) in the Cloud Adoption Framework.

Moving resources between subscriptions

As your subscription model grows, you might decide that some resources belong in other subscriptions. Many types of resources can be moved between subscriptions. You can also use automated deployments to re-create resources in another subscription. For more information, see [Move Azure resources to another resource group or subscription](#).

Tips for creating new subscriptions

- Identify who is responsible for creating new subscriptions.
- Decide which resource types are available in a subscription by default.
- Decide what all standard subscriptions should look like. Considerations include Azure RBAC access, policies, tags, and infrastructure resources.
- If possible, [programmatically create new subscriptions](#) via a service principal. You must [grant permission to the service principal](#) to create subscriptions. Define a security group that can request new subscriptions via an automated workflow.
- If you're an Enterprise Agreement (EA) customer, ask Azure Support to block creation of non-EA subscriptions for your organization.

Next steps

Create a management group hierarchy to help [organize and manage your subscriptions and resources](#).

[Organize and manage your subscriptions and resources](#)

Network topology and connectivity

Article • 06/13/2023

The network topology and connectivity design area are critical for establishing a foundation for your cloud network design.

Design area review

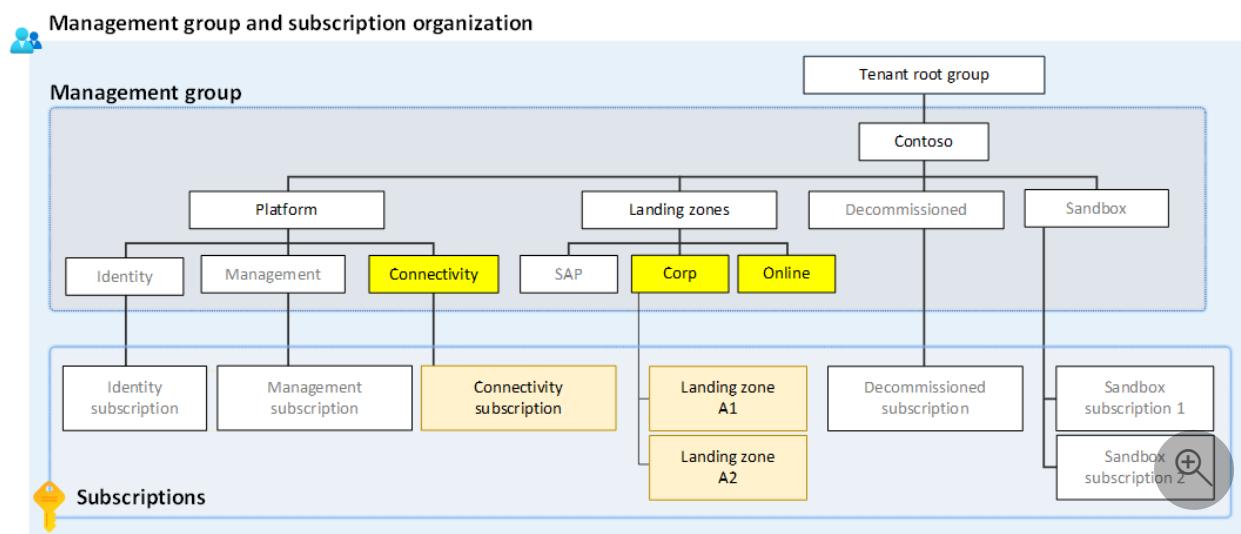
Involved roles or functions: This design area probably requires support from one or more [cloud platform](#) and [cloud center of excellence](#) functions to make and implement decisions.

Scope: The goal of network design is to align your cloud network design with overall cloud adoption plans. If your cloud adoption plans include hybrid or multicloud dependencies, or if you need connectivity for other reasons, your network design should also incorporate those connectivity options and expected traffic patterns.

Out of scope: This design area establishes the foundation for networking. It doesn't address compliance-related issues like advanced network security or automated enforcement guardrails. That guidance comes when you review the [security](#) and [governance](#) compliance design areas. Postponing security and governance discussions lets the cloud platform team address initial networking requirements before they expand their audience for more complex topics.

Design area overview

Network topology and connectivity are fundamental for organizations that are planning their landing zone design. Networking is central to almost everything inside a landing zone. It enables connectivity to other Azure services, external users, and on-premises infrastructure. Network topology and connectivity are in the [environmental group](#) of Azure landing zone design areas. This grouping is based on their importance in core design and implementation decisions.



In the [conceptual Azure landing zone architecture](#), there are two main management groups hosting workloads: Corp and Online. These management groups serve distinct purposes in organizing and governing Azure subscriptions. The networking relationship between the various Azure landing zones management groups depends on the organization's specific requirements and network architecture. The next few sections discuss the networking relationship between **Corp**, **Online**, and the **Connectivity** management groups in relation to what the Azure landing zone accelerator provides.

What is the purpose of Connectivity, Corp, and Online Management Groups?

- **Connectivity management group:** This management group contains dedicated subscriptions for connectivity, commonly a single subscription for most organizations. These subscriptions host the Azure networking resources required for the platform, like Azure Virtual WAN, Virtual Network Gateways, Azure Firewall, and Azure DNS private zones. It's also where hybrid connectivity is established between the cloud and on-premises environments, using services like ExpressRoute etc.
- **Corp management group:** The dedicated management group for corporate landing zones. This group is intended to contain subscriptions that host workloads that require traditional IP routing connectivity or hybrid connectivity with the corporate network via the hub in the connectivity subscription and therefore form part of the same routing domain. Workloads such as internal systems aren't exposed directly to the internet, but may be exposed via reverse proxies etc., such as Application Gateways.
- **Online management group:** The dedicated management group for online landing zones. This group is intended to contain subscriptions used for public-facing resources, such as websites, e-commerce applications, and customer-facing services. For example, organizations can use the Online management group to

isolate public-facing resources from the rest of the Azure environment, reducing the attack surface and ensuring that public-facing resources are secure and available to customers.

Why did we create Corp and Online management groups to separate workloads?

The difference in networking considerations between the Corp and Online management groups in the conceptual Azure landing zone architecture lies in their intended use and primary purpose.

The Corp management group is used to manage and secure internal resources and services, such as line-of-business applications, databases, and user management. The networking considerations for the Corp management group are focused on providing secure and efficient connectivity between internal resources, while enforcing strict security policies to protect against unauthorized access.

The Online management group in the conceptual Azure landing zone architecture can be considered as an isolated environment used to manage public-facing resources and services that are accessible from the Internet. By using the Online management group to manage public-facing resources, the Azure landing zone architecture provides a way to isolate those resources from internal resources, thereby reducing the risk of unauthorized access and minimizing the attack surface.

In the conceptual Azure landing zone architecture, the virtual network in the Online management group can be, optionally, peered with virtual networks in the Corp management group, either directly or indirectly via the hub and associated routing requirements via an Azure Firewall or NVA, allowing public-facing resources to communicate with internal resources in a secure and controlled manner. This topology ensures that the network traffic between public-facing resources and internal resources is secure and restricted, while still allowing the resources to communicate as needed.

💡 Tip

It is also important to understand and review the Azure Policies that are assigned, and inherited, on each of the Management Groups as part of the Azure landing zone. As these help shape, protect and govern the workloads that are deployed within the subscriptions that are in these Management Groups. The policy assignments for Azure landing zones can be found [here ↗](#).

Define an Azure network topology

Article • 05/29/2024

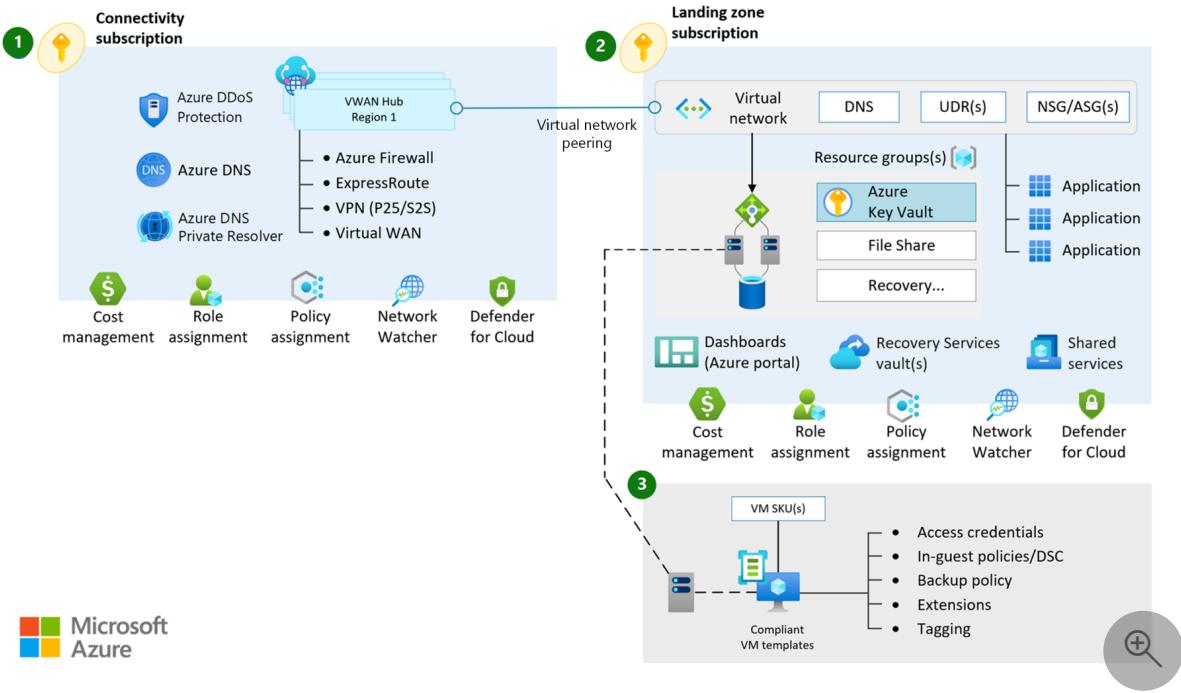
Network topology is a critical element of a landing zone architecture because it defines how applications can communicate with one another. This section explores technologies and topology approaches for Azure deployments. It focuses on two core approaches: topologies that are based on Azure Virtual WAN and traditional topologies.

Virtual WAN network topology

You can use [Virtual WAN to meet large-scale interconnectivity requirements](#). Virtual WAN is a service that Microsoft manages, which reduces overall network complexity and helps to modernize your organization's network. Use a Virtual WAN topology if any of the following requirements apply to your organization:

- Your organization intends to deploy resources across several Azure regions and requires global connectivity between virtual networks in these Azure regions and multiple on-premises locations.
- Your organization intends to use a software-defined WAN (SD-WAN) deployment to integrate a large-scale branch network directly into Azure, or requires more than 30 branch sites for native IPSec termination.
- You require transitive routing between a virtual private network (VPN) and Azure ExpressRoute. For example, if you use a site-to-site VPN to connect remote branches or a point-to-site VPN to connect remote users, you might need to connect the VPN to an ExpressRoute-connected DC through Azure.

The following diagram shows a Microsoft-managed Virtual WAN network topology:

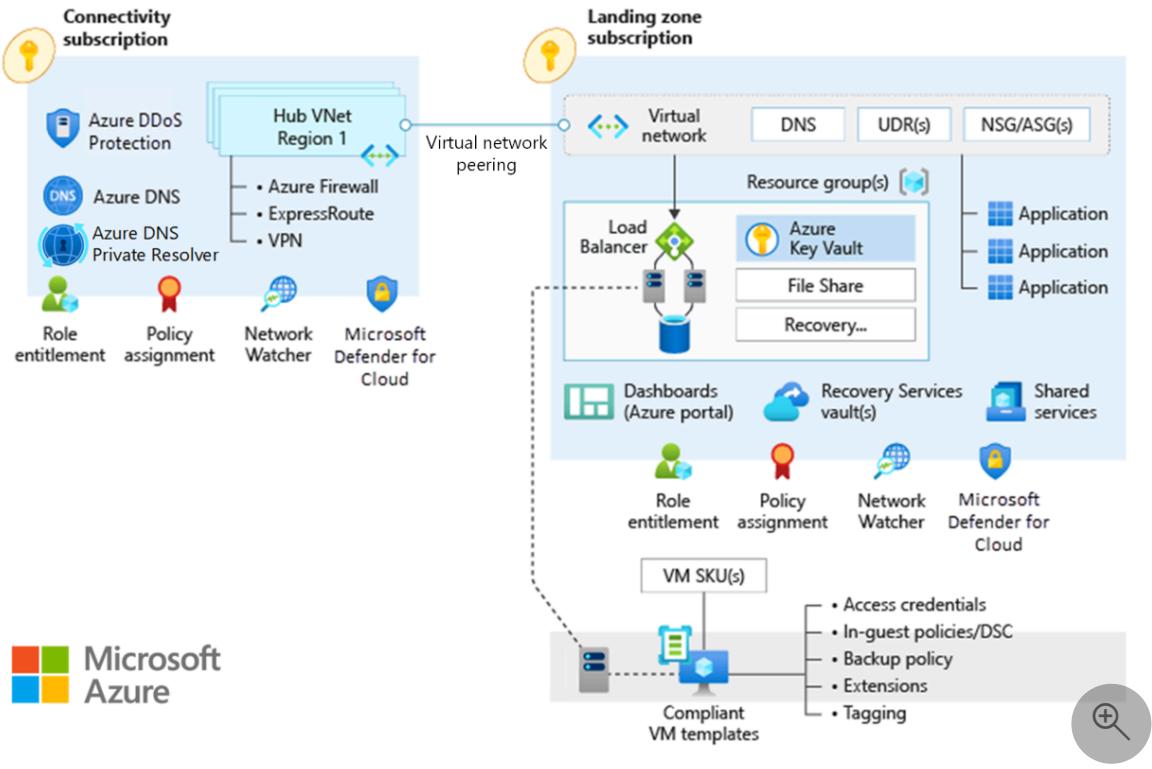


Traditional Azure networking topology

A [traditional hub-and-spoke network topology](#) helps you build customized, enhanced-security, large-scale networks in Azure. With this topology, you manage the routing and security. Use a traditional topology if any of the following requirements apply to your organization:

- Your organization intends to deploy resources across one or several Azure regions. You expect some traffic across Azure regions, such as traffic between two virtual networks across two different Azure regions, but you don't need a full-mesh network across all Azure regions.
- You have a low number of remote or branch locations for each region and require fewer than 30 IPSec site-to-site tunnels.
- You require full control and granularity to manually configure your Azure network routing policy.

The following diagram shows a traditional Azure networking topology:



Next step

[Virtual Network Manager in Azure landing zones](#)

Feedback

Was this page helpful?

Yes

No

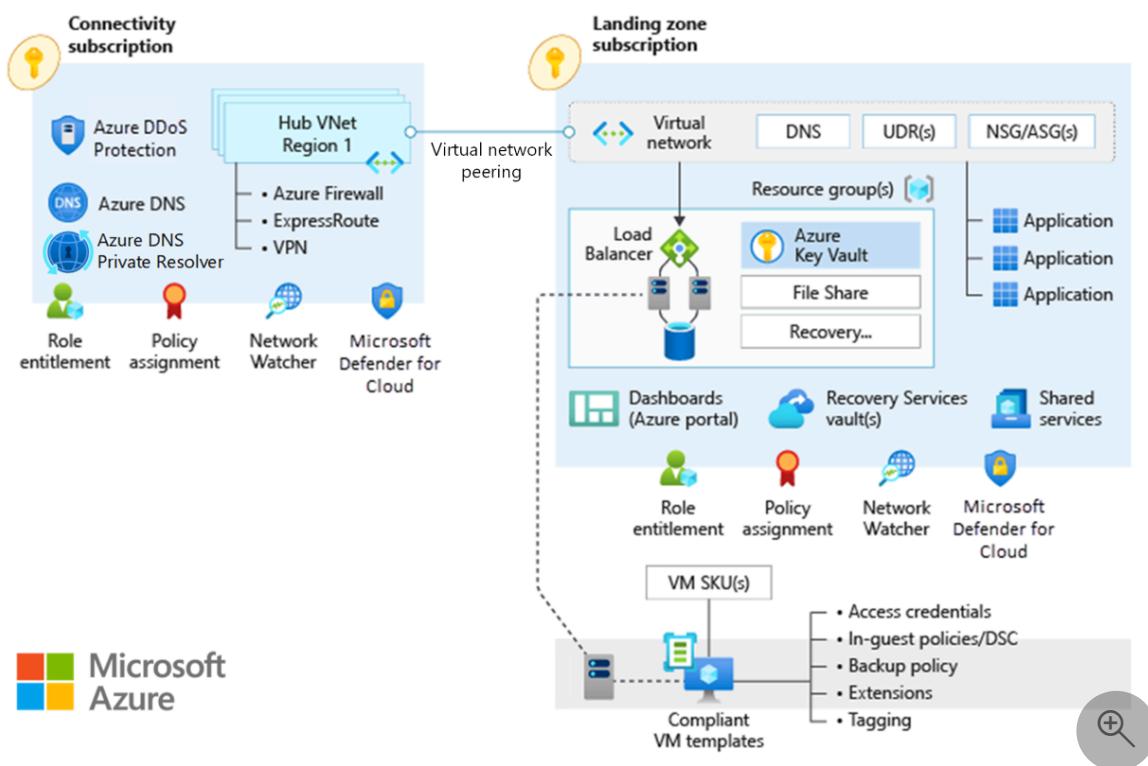
Traditional Azure networking topology

Article • 05/29/2024

ⓘ Important

Try the [topology \(preview\)](#) experience, which offers a visualization of Azure resources for ease of inventory management and monitoring network at scale. Use the topology feature in preview to visualize resources and their dependencies across subscriptions, regions, and locations. For more information about how to navigate to the experience, see [Azure Monitor](#).

This article describes key design considerations and recommendations for network topologies in Microsoft Azure. The following diagram shows a traditional Azure network topology:



Design considerations

- Various network topologies can connect multiple landing zone virtual networks. Examples of network topologies include hub-and-spoke, full-mesh, and hybrid topologies. You can also have multiple virtual networks that are connected via multiple Azure ExpressRoute circuits or connections.

- Virtual networks can't traverse subscription boundaries. However, you can use virtual network peering, an ExpressRoute circuit, or VPN gateways to achieve connectivity between virtual networks across different subscriptions.
- Virtual network peering is the preferred method to connect virtual networks in Azure. You can use virtual network peering to connect virtual networks in the same region, across different Azure regions, and across different Microsoft Entra tenants.
- Virtual network peering and global virtual network peering aren't transitive. To enable a transit network, you need user-defined routes (UDRs) and network virtual appliances (NVAs). For more information, see [Hub-spoke network topology in Azure](#).
- You can share an Azure DDoS Protection plan across all virtual networks in a single Microsoft Entra tenant to protect resources with public IP addresses. For more information, see [DDoS Protection](#).
 - DDoS Protection plans cover only resources with public IP addresses.
 - The cost of a DDoS Protection plan includes 100 public IP addresses across protected virtual networks that are associated with the DDoS Protection plan. Protection for more resources costs more. For more information, see [DDoS Protection pricing](#) or the [FAQ](#).
 - Review the [supported resources of DDoS Protection plans](#).
- You can use ExpressRoute circuits to establish connectivity across virtual networks within the same geopolitical region or use the premium add-on for connectivity across geopolitical regions. Keep these points in mind:
 - Network-to-network traffic might experience more latency, because traffic must hairpin at the Microsoft Enterprise edge (MSEE) routers.
 - The ExpressRoute gateway SKU constrains bandwidth.
 - Deploy and manage UDRs if you need to inspect or log UDRs for traffic across virtual networks.
- VPN gateways with Border Gateway Protocol (BGP) are transitive within Azure and on-premises networks, but they don't provide transitive access to networks connected through ExpressRoute by default. If you need transitive access to networks connected through ExpressRoute, consider [Azure Route Server](#).
- When you connect multiple ExpressRoute circuits to the same virtual network, use connection weights and BGP techniques to ensure an optimal path for traffic

between on-premises networks and Azure. For more information, see [Optimize ExpressRoute routing](#).

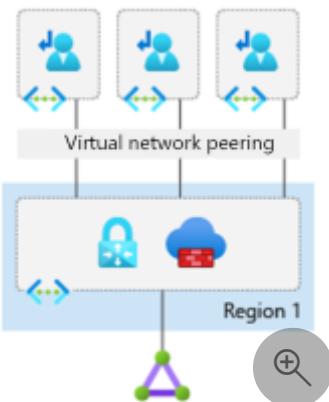
If you use BGP metrics to influence ExpressRoute routing, you need to change the configuration outside of the Azure platform. Your organization or your connectivity provider must configure the on-premises routers accordingly.

- ExpressRoute circuits with premium add-ons provide global connectivity.
- ExpressRoute has certain limits, including a maximum number of ExpressRoute connections for each ExpressRoute gateway. And ExpressRoute private peering has a maximum limit for the number of routes that it can identify from Azure to on-premises. For more information, see [ExpressRoute limits](#).
- A VPN gateway's maximum aggregated throughput is 10 gigabits per second. A VPN gateway supports up to 100 site-to-site or network-to-network tunnels.
- If an NVA is part of the architecture, consider Route Server to simplify dynamic routing between your NVA and your virtual network. Use Route Server to exchange routing information directly through BGP between any NVA that supports BGP and the Azure software-defined network (SDN) in the Azure virtual network. You don't need to manually configure or maintain route tables with this approach.

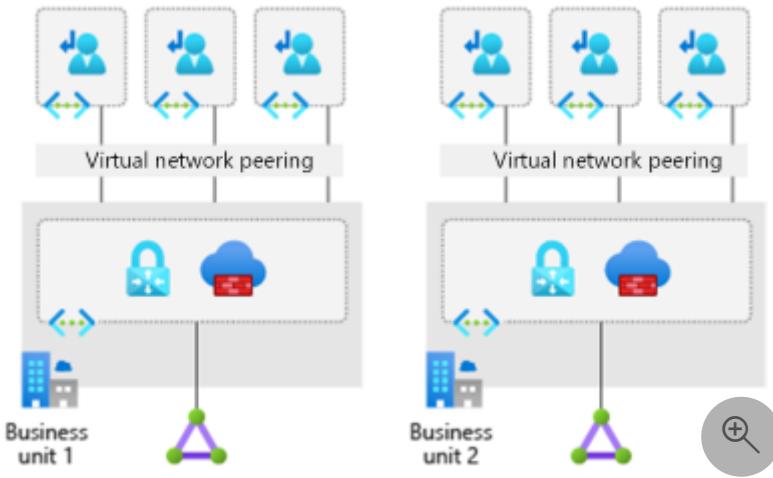
Design recommendations

- Consider a network design based on the traditional hub-and-spoke network topology for the following scenarios:
 - A network architecture deployed within a single Azure region.
 - A network architecture that spans multiple Azure regions, with no need for transitive connectivity between virtual networks for landing zones across regions.
 - A network architecture that spans multiple Azure regions, and global virtual network peering that can connect virtual networks across Azure regions.
 - There's no need for transitive connectivity between VPN and ExpressRoute connections.
 - The main hybrid connectivity method in place is ExpressRoute, and the number of VPN connections is less than 100 per VPN gateway.
 - There's a dependency on centralized NVAs and granular routing.

- For regional deployments, primarily use the hub-and-spoke topology with a regional hub for each spoke Azure region. Use application landing zone virtual networks that use virtual network peering to connect to a regional central hub virtual network for the following scenarios:
 - Cross-premises connectivity through ExpressRoute that's enabled in two different peering locations. For more information, see [Design and architect ExpressRoute for resiliency](#).
 - A VPN for branch connectivity.
 - Spoke-to-spoke connectivity through NVAs and UDRs.
 - Internet-outbound protection through Azure Firewall or another non-Microsoft NVA.
- The following diagram shows the hub-and-spoke topology. Use this configuration to ensure appropriate traffic control and to meet most requirements for segmentation and inspection.

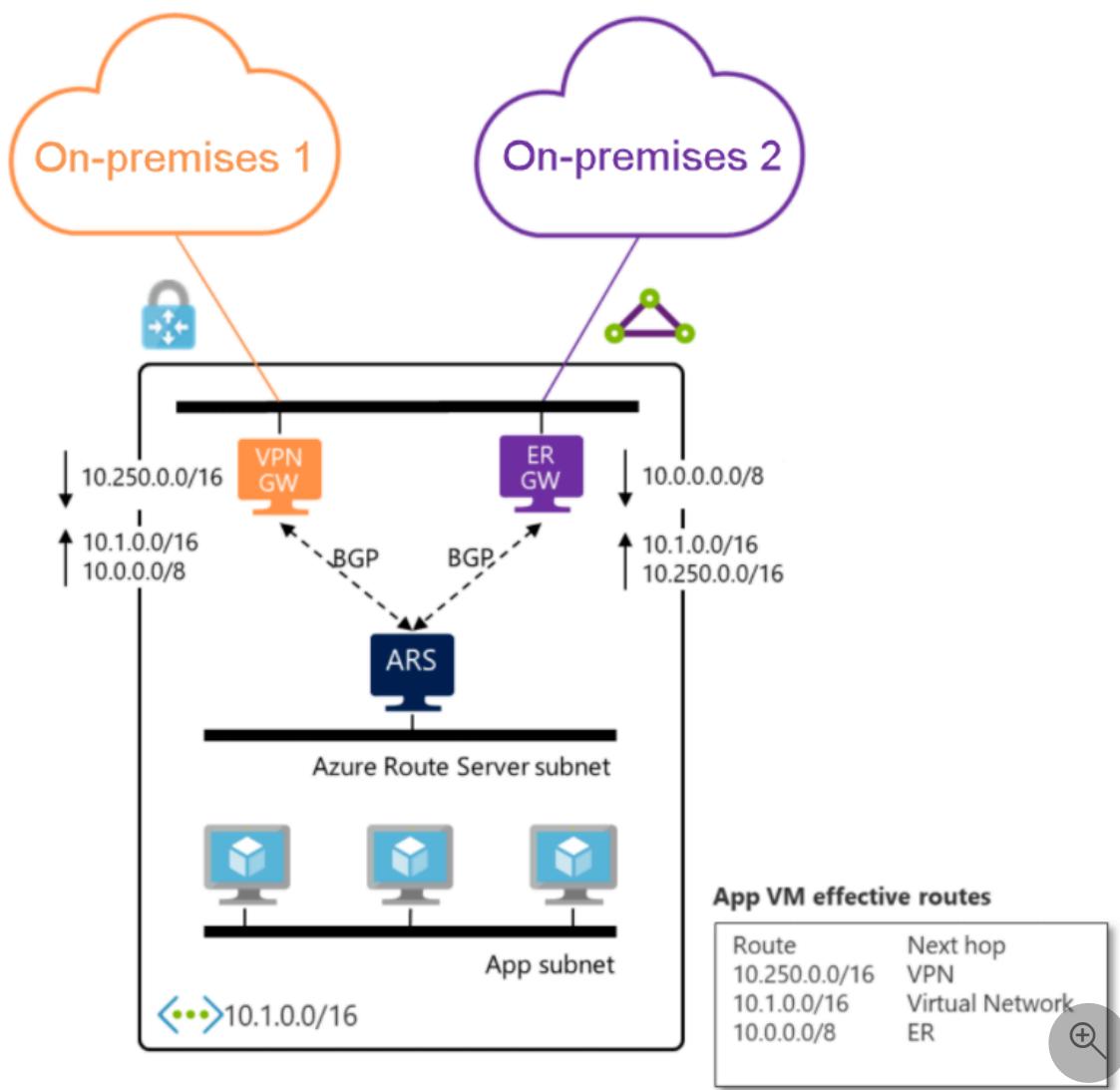


- Use the topology that has multiple virtual networks that are connected via multiple ExpressRoute circuits at different peering locations if:
 - You need a high level of isolation. For more information, see [Design and architect ExpressRoute for resiliency](#).
 - You need dedicated ExpressRoute bandwidth for specific business units.
 - You reach the maximum number of connections for each ExpressRoute gateway. To determine the maximum number, see [ExpressRoute limits](#).
- The following diagram shows this topology.



- Deploy Azure Firewall or partner NVAs in the central-hub virtual network for east/west or south/north traffic protection and filtering.
- Deploy a single DDoS Protection standard plan in the connectivity subscription. Use this plan for all landing zone and platform virtual networks.
- Use your existing network, multiprotocol label switching (MPLS), and SD-WAN to connect branch locations with corporate headquarters. If you don't use Route Server, then you don't have support for transit in Azure between ExpressRoute connections and VPN gateways.
- Deploy Azure Firewall or partner NVAs for east/west or south/north traffic protection and filtering, in the central-hub virtual network.
- When you deploy partner networking technologies or NVAs, follow the partner vendor's guidance to ensure that:
 - The vendor supports deployment.
 - The guidance supports high availability and maximum performance.
 - There are no conflicting configurations with Azure networking.
- Don't deploy Layer 7 inbound NVAs, such as Azure Application Gateway, as a shared service in the central-hub virtual network. Instead, deploy them together with the application in their respective landing zones.
- Deploy a single DDoS standard protection plan in the connectivity subscription.
 - All landing zone and platform virtual networks should use this plan.
- Use your existing network, multiprotocol label switching, and SD-WAN to connect branch locations with corporate headquarters. If you don't use Route Server, then there's no support for transit in Azure between ExpressRoute and VPN gateways.

- If you need transitivity between ExpressRoute and VPN gateways in a hub-and-spoke scenario, use Route Server. For more information, see [Route Server support for ExpressRoute and Azure VPN](#).

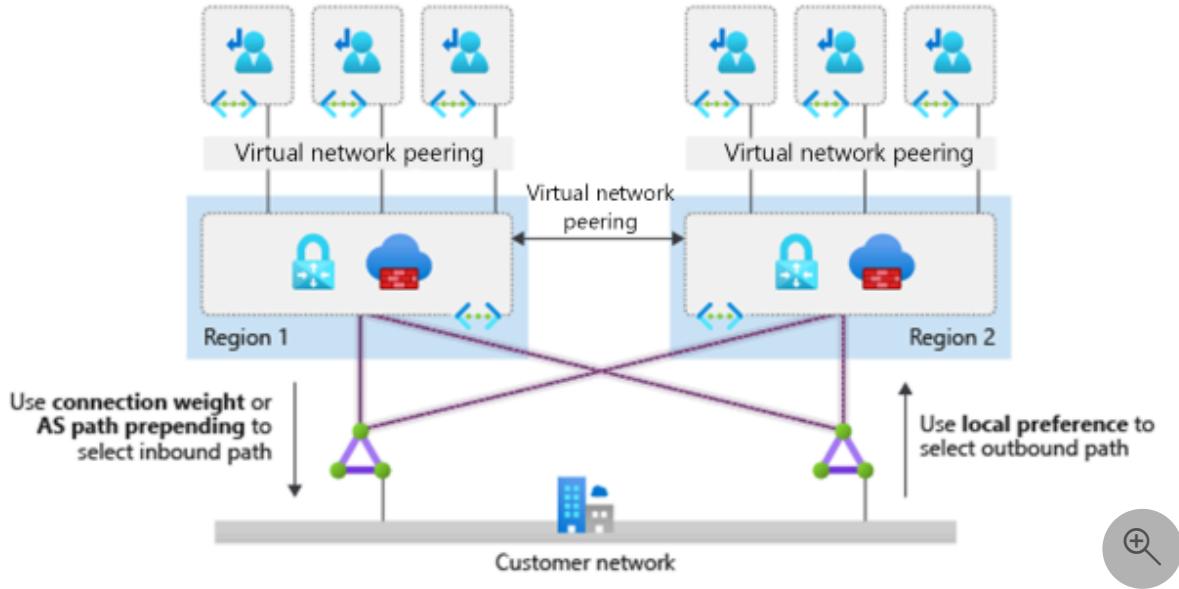


- When you have hub-and-spoke networks in multiple Azure regions, and you need to connect a few landing zones across regions, use global virtual network peering. You can directly connect landing zone virtual networks that need to route traffic to each other. Depending on the communicating virtual machine's SKU, global virtual network peering can provide high network throughput. Traffic that goes between directly peered landing zone virtual networks bypasses NVAs within hub virtual networks. Limitations on global virtual network peering apply to the traffic.
- When you have hub-and-spoke networks in multiple Azure regions, and you need to connect most landing zones across regions, use hub NVAs to connect hub virtual networks in each region to each other and to route traffic across regions. You can also use this approach if you can't use direct peering to bypass hub NVAs because of incompatibility with your security requirements. Global virtual network

peering or ExpressRoute circuits can help to connect hub virtual networks in the following ways:

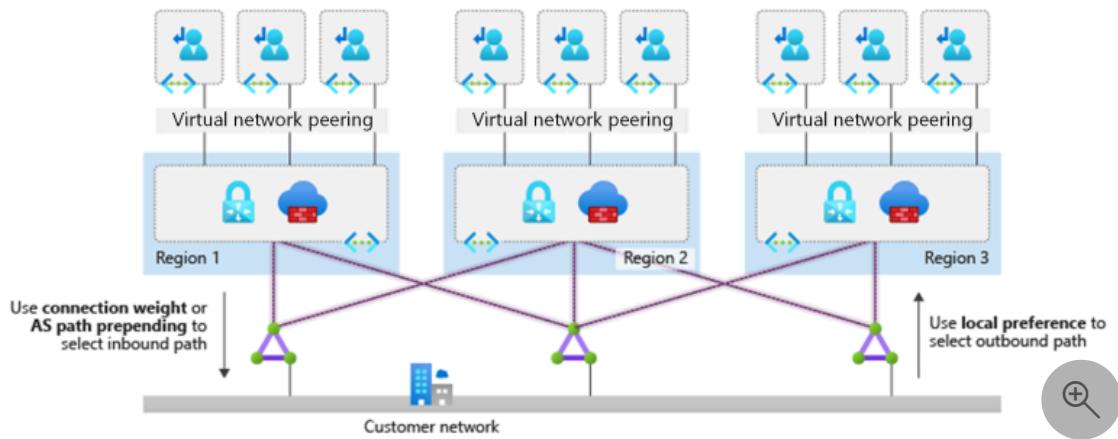
- Global virtual network peering provides a low latency and high throughput connection but generates [traffic fees](#).
- If you route through ExpressRoute, you might increase latency due to the MSEE hairpin. The selected [ExpressRoute gateway SKU](#) limits the throughput.

The following diagram shows options for hub-to-hub connectivity:



- When you need to connect two Azure regions, use global virtual network peering to connect the hub virtual networks in each region.
- Use a managed global transit network architecture that's based on [Azure Virtual WAN](#) if your organization:
 - Requires hub-and-spoke network architectures across more than two Azure regions.
 - Requires global transit connectivity between landing zones virtual networks across Azure regions.
 - Wants to minimize network management overhead.
- When you need to connect more than two Azure regions, then we recommend that the hub virtual networks in each region connect to the same ExpressRoute circuits. Global virtual network peering requires you to manage a large number of peering relationships and a complex set of UDRs across multiple virtual networks.

The following diagram shows how to connect hub-and-spoke networks in three regions:



- When you use ExpressRoute circuits for cross-region connectivity, spokes in different regions communicate directly and bypass the firewall because they learn through BGP routes to the spokes of the remote hub. If you need the firewall NVAs in the hub virtual networks to inspect traffic across spokes, you must implement one of these options:
 - Create more specific route entries in the spoke UDRs for the firewall in the local hub virtual network to redirect traffic across hubs.
 - To simplify route configuration, [disable BGP propagation](#) on the spoke route tables.
- When your organization requires hub-and-spoke network architectures across more than two Azure regions and global transit connectivity between landing zones virtual networks across Azure regions, and you want to minimize network management overhead, we recommend a managed global transit network architecture that's based on [Virtual WAN](#).
- Deploy each region's hub network resources into separate resource groups, and sort them into each deployed region.
- Use [Azure Virtual Network Manager](#) to manage connectivity and security configuration of virtual networks globally across subscriptions.
- Use [Azure Monitor network insights](#) to monitor the end-to-end state of your networks on Azure.
- You must consider the following two [limits](#) when you connect spoke virtual networks to the central hub virtual network:

- The maximum number of virtual network peering connections per virtual network.
- The maximum number of prefixes that ExpressRoute with private peering advertises from Azure to on-premises.
- Ensure that the number of spoke virtual networks connected to the hub virtual network don't exceed these limits.

Next step

[Virtual WAN network topology](#)

Feedback

Was this page helpful?

 Yes

 No

Virtual WAN network topology

Article • 08/01/2024

Explore key design considerations and recommendations for virtual wide area networks (Virtual WAN) in Microsoft Azure.

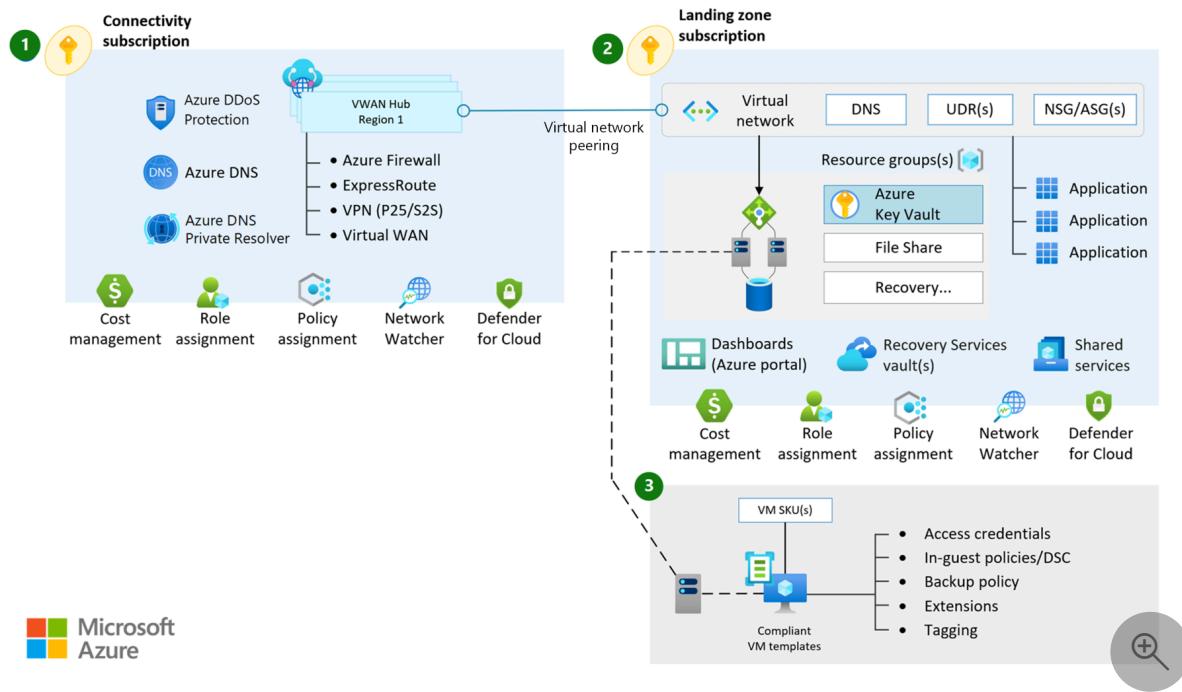


Figure 1: Virtual WAN network topology. Download a [Visio file](#) of this architecture.

Virtual WAN network design considerations

Azure Virtual WAN is a Microsoft-managed solution that provides end-to-end, global, and dynamic transit connectivity by default. Virtual WAN hubs eliminate the need to manually configure network connectivity. For example, you don't need to manage user-defined routes (UDR) or network virtual appliances (NVAs) to enable global transit connectivity.

- Azure Virtual WAN simplifies end-to-end network connectivity in Azure, and to Azure from on-premises, by creating a [hub-and-spoke network architecture](#). The architecture easily scales to support multiple Azure regions and on-premises locations (any-to-any connectivity) as shown in the following figure:

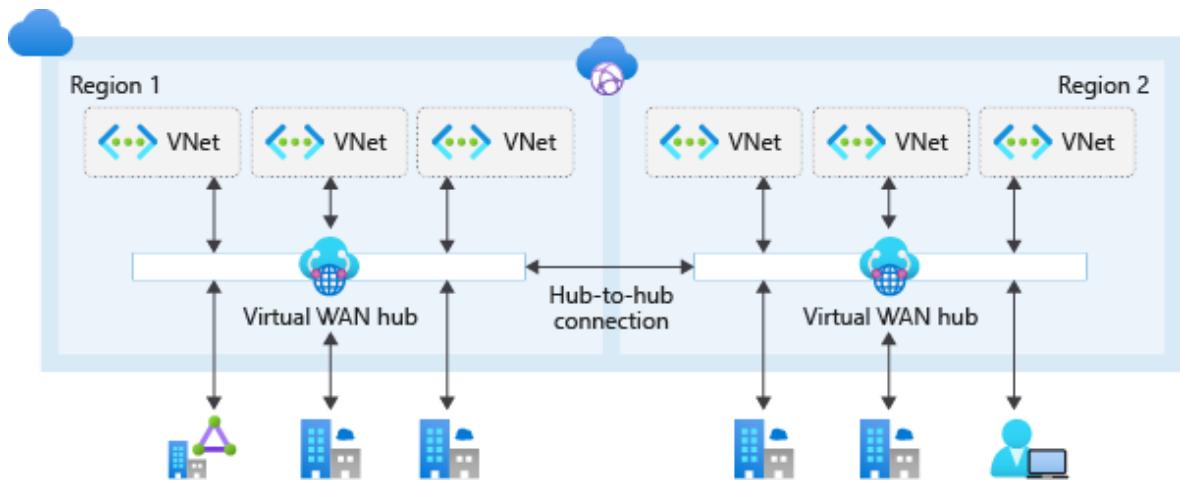


Figure 2: Global transit network with Virtual WAN.

- Azure Virtual WAN any-to-any transitive connectivity supports the following paths (within the same region and across regions):
 - Virtual network to virtual network
 - Virtual network to branch
 - Branch to virtual network
 - Branch to branch
- Azure Virtual WAN hubs are restricted to the deployment of Microsoft managed resources. The only resources that you can deploy within the WAN hubs are:
 - Virtual network gateways (point-to-site VPN, site-to-site VPN, and Azure ExpressRoute)
 - Azure Firewall via Firewall Manager
 - Route tables
 - Some [network virtual appliances \(NVA\)](#) for vendor-specific SD-WAN capabilities
- Virtual WAN is bound by [Azure subscription limits for Virtual WAN](#).
- Network-to-network transitive connectivity (within a region and across regions via hub-to-hub) is in general availability (GA).
- The Microsoft-managed routing function that's a part of every virtual hub enables the transit connectivity between virtual networks in Standard Virtual WAN. Each hub supports an aggregate throughput of up to 50 Gbps for VNet-to-VNet traffic.
- A single Azure Virtual WAN hub supports a specific maximum number of VM workloads across all directly attached VNets. For more information, see [Azure Virtual WAN limits](#).
- You can deploy multiple Azure Virtual WAN hubs in the same region to scale beyond the single hub limits.

- Virtual WAN integrates with various [SD-WAN providers](#).
- Many managed service providers offer [managed services](#) for Virtual WAN.
- User VPN (point-to-site) gateways in Virtual WAN scale up to 20-Gbps aggregated throughput and 100,000 client connections per virtual hub. For more information, see [Azure Virtual WAN limits](#).
- Site-to-site VPN gateways in Virtual WAN scale up to 20-Gbps aggregated throughput.
- You can connect ExpressRoute circuits to a Virtual WAN hub by using a Local, Standard, or Premium SKU.
- For deployments in the same city, consider [ExpressRoute Metro](#).
- ExpressRoute Standard or Premium circuits, in locations supported by Azure ExpressRoute Global Reach, can connect to a Virtual WAN ExpressRoute gateway. And they have all the Virtual WAN transit capabilities (VPN-to-VPN, VPN, and ExpressRoute transit). ExpressRoute Standard or Premium circuits that are in locations not supported by Global Reach can connect to Azure resources, but can't use Virtual WAN transit capabilities.
- Azure Firewall Manager supports deployment of Azure Firewall in the Virtual WAN hub, known as secured virtual hub. For more information, see the [Azure Firewall Manager overview](#) for secured virtual hubs and the latest [constraints](#).
- Virtual WAN hub-to-hub traffic that goes through Azure Firewall in both source hubs and target hubs (secured virtual hubs) is supported when you enable routing intent and policies. For more information, see [Use cases for Virtual WAN hub routing intent and routing policies](#).
- The Virtual WAN portal experience requires that all Virtual WAN resources deploy together into the same resource group.
- You can share an Azure DDoS Protection plan across all VNets in a single Microsoft Entra tenant to protect resources with public IP addresses. For more information, see [Azure DDoS Protection](#).
 - Virtual WAN secure virtual hubs don't support Azure DDoS standard protection plans. For more information, see [Azure Firewall Manager known issues](#) and [Hub virtual network and secured virtual hub comparison](#).
 - Azure DDoS Protection plans only cover resources with public IP addresses.

- An Azure DDoS Protection plan includes 100 public IP addresses. These public IP addresses span all protected VNets associated with the DDoS protection plan. Any other public IP addresses, beyond the 100 included with the plan, are charged separately. For more information on Azure DDoS Protection pricing, see the [pricing page](#) or the [FAQ](#).
- Review the [supported resources of Azure DDoS Protection plans](#).

Virtual WAN network design recommendations

We recommend Virtual WAN for new large or global network deployments in Azure where you need global transit connectivity across Azure regions and on-premises locations. That way, you don't have to manually set up transitive routing for Azure networking.

The following figure shows a sample global enterprise deployment with datacenters spread across Europe and the United States. The deployment contains many branch offices within both regions. The environment is globally connected via Azure Virtual WAN and [ExpressRoute Global Reach](#).

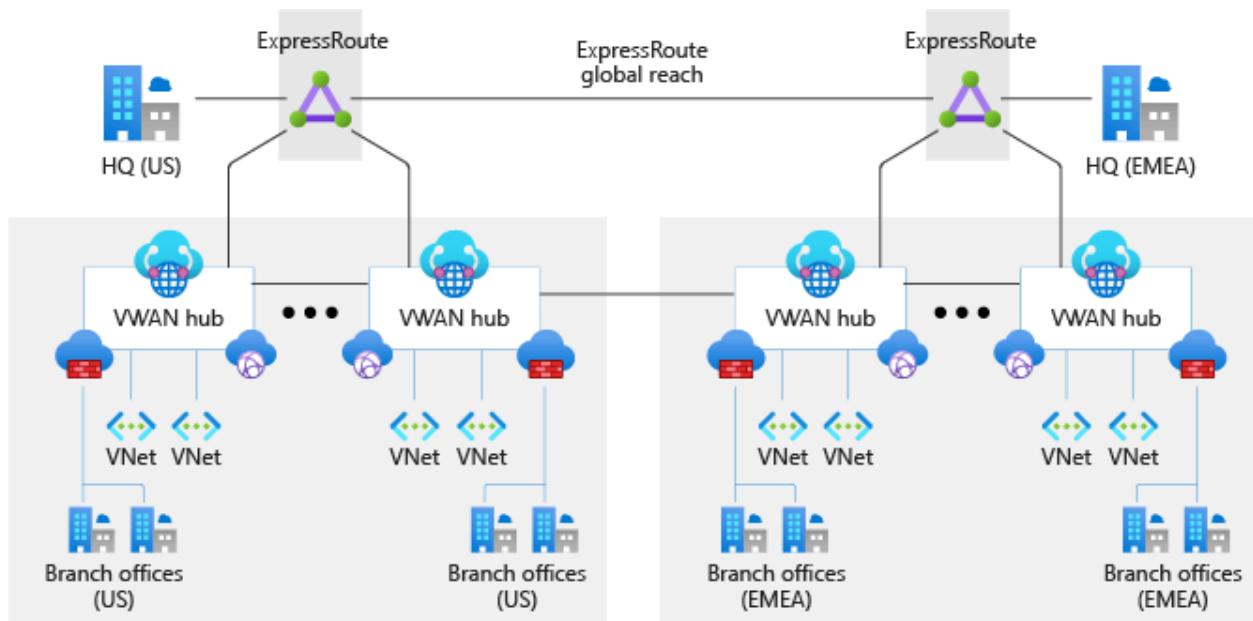


Figure 3: Sample network topology.

- Use a Virtual WAN hub per Azure region to connect multiple landing zones together across Azure regions by way of a common global Azure Virtual WAN.
- Deploy all Virtual WAN resources into a single resource group in the connectivity subscription, including when you're deploying across multiple regions.

- Use [virtual hub routing](#) features to further segment traffic between VNets and branches.
- Connect Virtual WAN hubs to on-premises datacenters by using ExpressRoute.
- Deploy required shared services, like DNS servers, in a dedicated spoke virtual network. Customer deployed shared resources can't be deployed inside the Virtual WAN hub itself.
- Connect branches and remote locations to the nearest Virtual WAN hub via Site-to-Site VPN, or enable branch connectivity to Virtual WAN via an SD-WAN partner solution.
- Connect users to the Virtual WAN hub via a Point-to-Site VPN.
- Follow the principle of "traffic in Azure stays in Azure" so that communication across resources in Azure occurs via the Microsoft backbone network, even when the resources are in different regions.
- For internet outbound protection and filtering, consider deploying Azure Firewall in the virtual hub.
- [Security provided by NVA firewalls](#). Customers can also deploy NVAs into a Virtual WAN hub that performs both SD-WAN connectivity and Next-Generation Firewall capabilities. Customers can connect on-premises devices to the NVA in the hub and also use the same appliance to inspect all North-South, East-West, and Internet-bound traffic.
- When you're deploying partner networking technologies and NVAs, follow the partner vendor's guidance to ensure there are no conflicting configurations with Azure networking.
- For brownfield scenarios where you're migrating from a hub-and-spoke network topology not based on Virtual WAN, see [Migrate to Azure Virtual WAN](#).
- Create Azure Virtual WAN and Azure Firewall resources within the connectivity subscription.
- Use [Virtual WAN hub routing intent and routing policies](#) to support traffic that goes between secured hubs.
- Don't create more than 500 virtual network connections per Virtual WAN virtual hub.
 - If you need more than 500 virtual network connections per Virtual WAN virtual hub, you can deploy another Virtual WAN virtual hub. Deploy it in the same

region as part of the same Virtual WAN and resource group.

- Plan your deployment carefully, and ensure that your network architecture is within the [Azure Virtual WAN limits](#).
 - Use [insights in Azure Monitor for Virtual WAN \(preview\)](#) to monitor the end-to-end topology of your Virtual WAN and status and [key metrics](#).
 - Deploy a single Azure DDoS standard protection plan in the connectivity subscription.
 - All landing zone and platform VNets should use this plan.
-

Feedback

Was this page helpful?

 Yes

 No

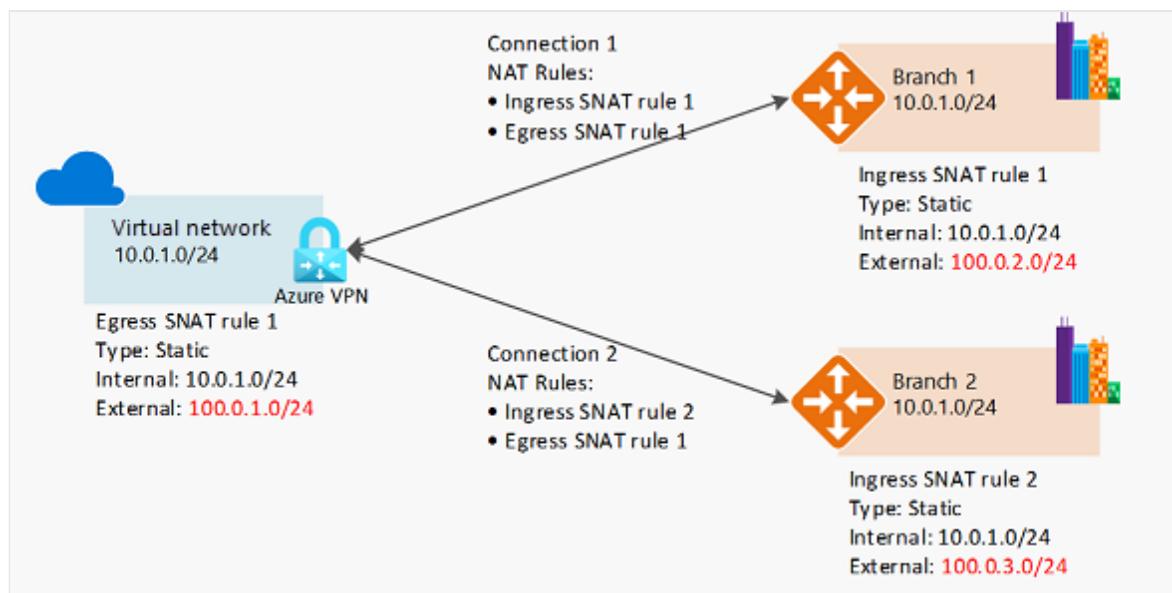
Plan for IP addressing

Article • 01/10/2024

It's important your organization plans for IP addressing in Azure. Planning ensures the IP address space doesn't overlap across on-premises locations and Azure regions.

Design considerations:

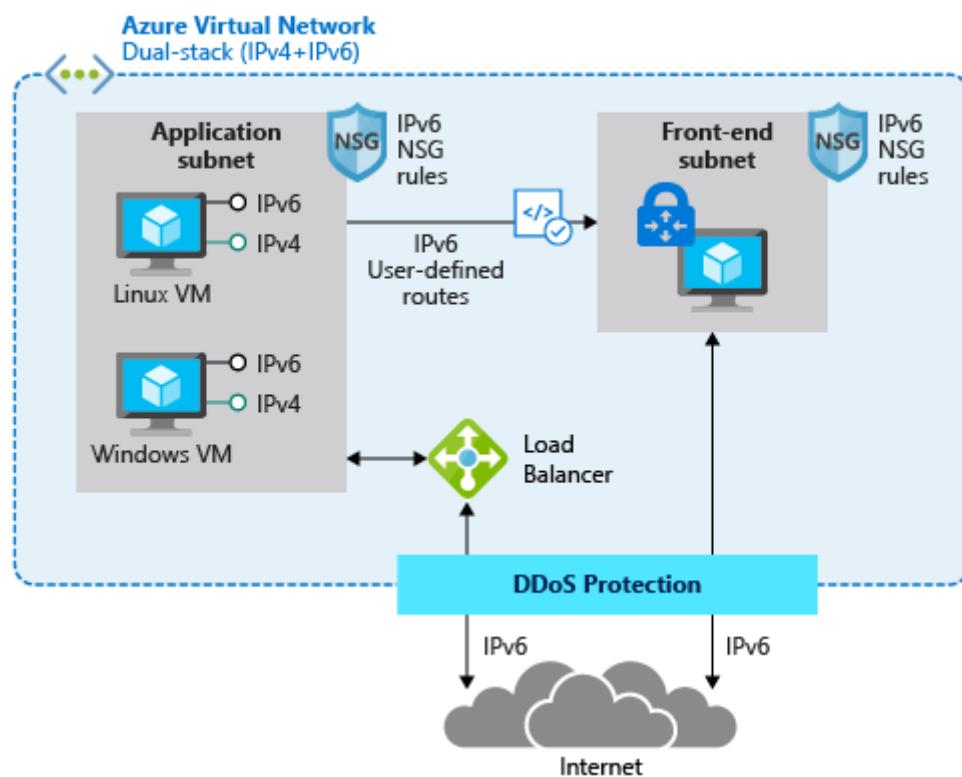
- Overlapping IP address spaces across on-premises and Azure regions creates major contention challenges.
- Azure VPN Gateway can connect overlapping, on-premises sites with overlapping IP address spaces through network address translation (NAT) capability. This feature is generally available in [Azure Virtual WAN](#) and standalone [Azure VPN Gateway](#).



- You can add address space after you create a virtual network. This process doesn't need an outage if the virtual network is already connected to another virtual network via virtual network peering. Instead, each remote peering needs a [resync operation](#) done after the network space has changed.
- Azure reserves five IP addresses within each subnet. Factor in those addresses when you're sizing virtual networks and encompassed subnets.
- Some Azure services require [dedicated subnets](#). These services include Azure Firewall and Azure VPN Gateway.
- You can delegate subnets to certain services to create instances of a service within the subnet.

Design recommendations:

- Plan for non-overlapping IP address spaces across Azure regions and on-premises locations in advance.
- Use IP addresses from the address allocation for private internet, known as RFC 1918 addresses.
- Don't use the following address ranges:
 - 224.0.0.0/4 (multicast)
 - 255.255.255.255/32 (broadcast)
 - 127.0.0.0/8 (loopback)
 - 169.254.0.0/16 (link-local)
 - 168.63.129.16/32 (internal DNS)
- For environments that have limited availability of private IP addresses, consider using IPv6. Virtual networks can be IPv4-only or dual stack [IPv4+IPv6](#).



- Don't create large virtual networks like /16. It ensures that IP address space isn't wasted. The smallest supported IPv4 subnet is /29, and the largest is /2 when using classless inter-domain routing (CIDR) subnet definitions. IPv6 subnets must be exactly /64 in size.
- Don't create virtual networks without planning the required address space in advance.

- Don't use public IP addresses for virtual networks, especially if the public IP addresses don't belong to your organization.
- Take the services you're going to use into consideration, there are some services with reserved IPs (IP Addresses), like [AKS with CNI networking](#)
- Use [nonroutable landing zone spoke virtual networks](#) and [Azure Private Link service](#) to prevent IPv4 exhaustion.

IPv6 considerations

An increasing number of organizations are adopting IPv6 in their environments. This adoption is driven by the public IPv4 space exhaustion, private IPv4 scarcity, especially within large-scale networks, and the need to provide connectivity to IPv6-only clients. There's no universal approach to adopting IPv6. There are, however, best practices that you can follow when you plan for IPv6 and implement it in your existing Azure networks.

The Microsoft [Cloud Adoption Framework](#) for Azure helps you understand the considerations to take into account when you create systems in the cloud. To learn about architectural best practices for designing sustainable systems, see [Azure landing zone design principles](#). For in-depth recommendations and best practices regarding your cloud architecture, including reference architecture deployments, diagrams, and guides, see the [Azure Architecture Center](#).

Design considerations:

- Phase your IPv6 adoption. Based on your business needs, implement IPv6 where needed. Remember that IPv4 and IPv6 can coexist as long as necessary.
- In scenarios where applications rely on infrastructure as a service (IaaS) services that have full IPv6 support, like virtual machines (VMs), native end-to-end use of IPv4 and IPv6 is possible. This configuration avoids translation complications and provides the most information to the server and application.

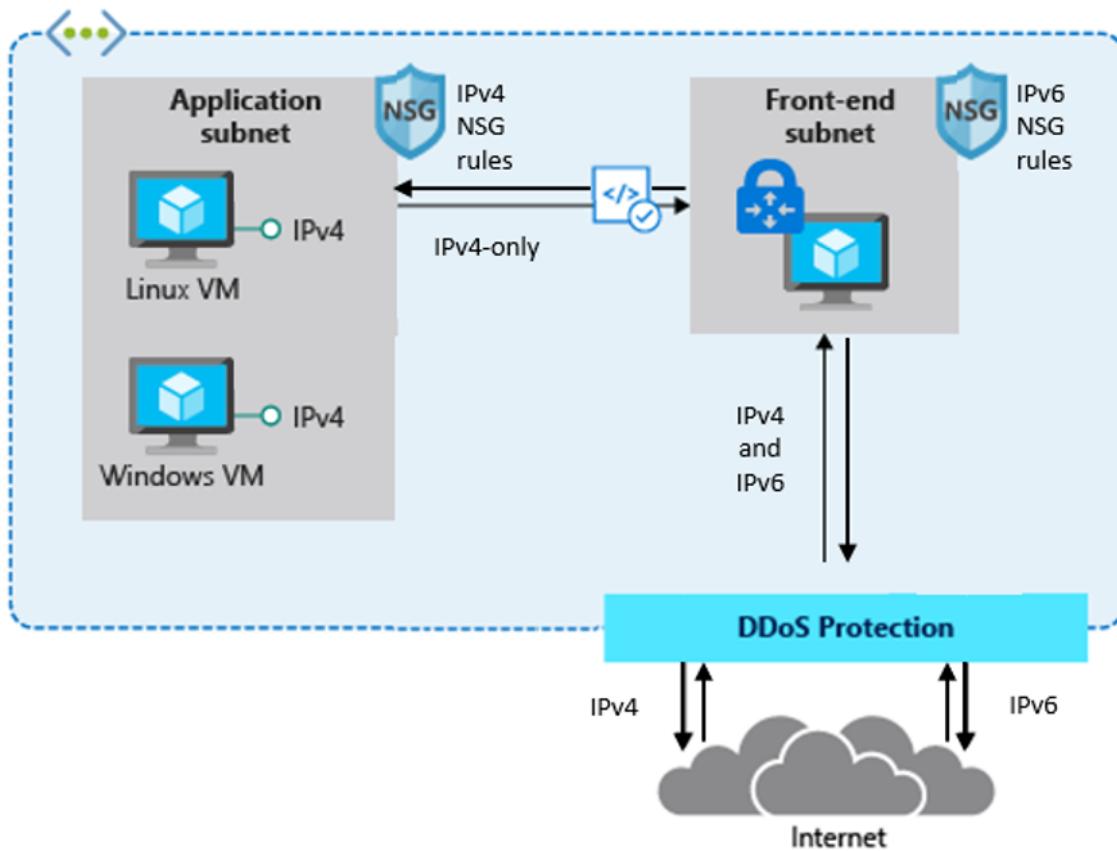
You can deploy Basic-SKU internet-facing Azure load balancers with an IPv6 address. This configuration enables native end-to-end IPv6 connectivity between the public internet and Azure VMs via the load balancer. This approach also facilitates native end-to-end outbound connections between VMs and IPv6-enabled clients on the public internet. Note that this approach requires every device in the path to handle IPv6 traffic.

The native end-to-end approach is most useful for direct server-to-server or client-to-server communication. It's not useful for most web services and applications,

which are typically protected by firewalls, web application firewalls, or reverse proxies.

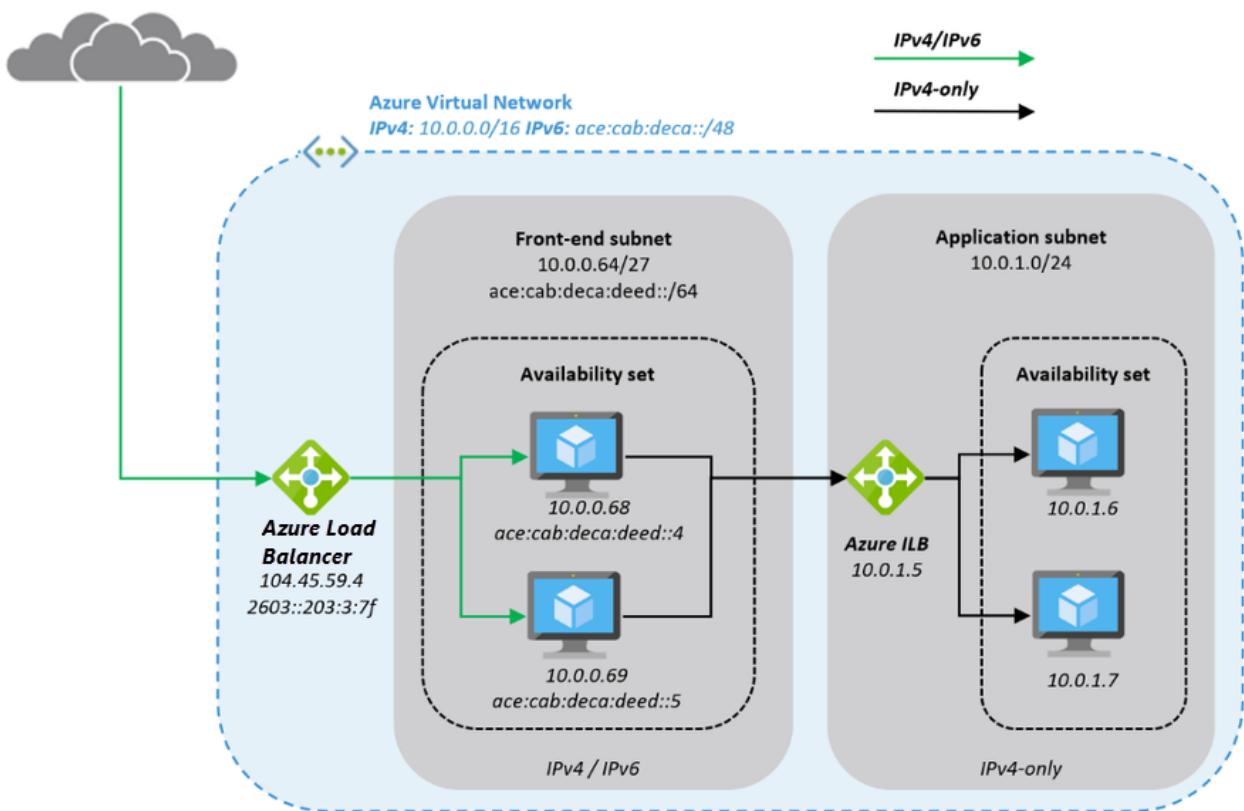
- Some complex deployments and applications that use a combination of third-party services, platform as a service (PaaS) services, and back-end solutions might not support native IPv6. In these cases, you need to use NAT/NAT64 or an IPv6 proxy solution to enable communication between IPv6 and IPv4.
- When the complexity of the application architecture or other factors like training costs are considered significant, you might want to keep using IPv4-only infrastructure on the back end and deploy a third-party network virtual appliance (NVA) dual-stack IPv4/IPv6 gateway for service delivery.

A typical deployment that uses an NVA might look like this:



Design recommendations:

Here's a closer look at what a typical architecture might look like:

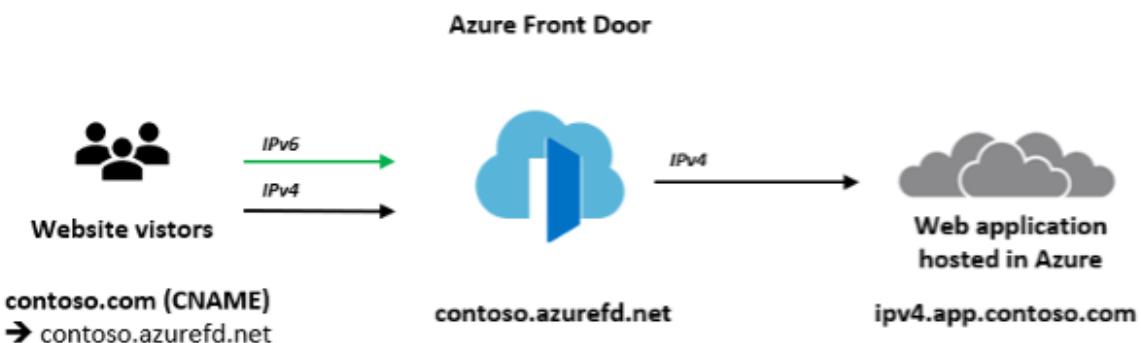


- Deploy the NVA in Azure availability sets for resiliency and expose them to the internet through [Azure Load-Balancer](#), which has a public IP address front end.

The NVAs accept IPv4 and IPv6 traffic and translate it into IPv4-only traffic to access the application in the application subnet. The approach reduces complexity for the application team and reduces the attack surface.

- Deploy [Azure Front Door](#) to provide global routing for web traffic.

Azure Front Door capabilities include proxying IPv6 client requests and traffic to an IPv4-only back end, as shown here:



These are main differences between the NVA approach and the Azure Front Door approach:

- NVAs are customer-managed, work at Layer 4 of the OSI model, and can be deployed in the same Azure virtual network as the application, with a private and

public interface.

- Azure Front Door is a global Azure PaaS service and operates at Layer 7 (HTTP/HTTPS). The application back end is an internet-facing service that can be locked down to accept only traffic from Azure Front Door.

In complex environments, you can use a combination of both. NVAs are used within a regional deployment. Azure Front Door is used to route traffic to one or more regional deployments in different Azure regions or other internet-facing locations. To determine the best solution, we recommend that you review the capabilities of [Azure Front Door](#) and the product documentation.

IPv6 virtual network CIDR blocks:

- You can associate a single IPv6 Classless Inter-Domain Routing (CIDR) block when you create a new virtual network in an existing Azure deployment in your subscription. The size of the subnet for IPv6 must be /64. Using this size ensures future compatibility if you decide to enable routing of the subnet to an on-premises network. Some routers can accept only /64 IPv6 routes.
- If you have an existing virtual network that supports only IPv4, and resources in your subnet that are configured to use only IPv4, you can enable IPv6 support for your virtual network and resources. Your virtual network can operate in dual-stack mode, which enables your resources to communicate over IPv4, IPv6, or both. IPv4 and IPv6 communication are independent of each other.
- You can't disable IPv4 support for your virtual network and subnets. IPv4 is the default IP addressing system for Azure virtual networks.
- Associate an IPv6 CIDR block with your virtual network and subnet or BYOIP IPv6. CIDR notation is a method of representing an IP address and its network mask. The formats of these addresses are as follows:
 - An individual IPv4 address is 32 bits, with four groups of as many as three decimal digits. For example, `10.0.1.0`.
 - An IPv4 CIDR block has four groups of as many as three decimal digits, from 0 through 255, separated by periods, and followed by a slash and a number from 0 through 32. For example, `10.0.0.0/16`.
 - An individual IPv6 address is 128 bits. It has eight groups of four hexadecimal digits. For example, `2001:0db8:85a3:0000:0000:8a2e:0370:7334`.
 - An IPv6 CIDR block has four groups of as many as four hexadecimal digits, separated by colons, followed by a double colon, and then followed by a slash and a number from 1 through 128. For example, `2001:db8:1234:1a00::/64`.
- Update your route tables to route IPv6 traffic. For public traffic, create a route that routes all IPv6 traffic from the subnet to VPN Gateway or an Azure ExpressRoute gateway.

- Update your security group rules to include rules for IPv6 addresses. Doing so enables IPv6 traffic to flow to and from your instances. If you have network security group rules to control the flow of traffic to and from your subnet, you must include rules for IPv6 traffic.
- If your instance type doesn't support IPv6, use dual stack or deploy an NVA, as previously described, that translates from IPv4 to IPv6.

IP Address Management (IPAM) tools

Using an IPAM tool can assist you with IP address planning in Azure as it provides centralized management and visibility, preventing overlaps and conflicts in IP address spaces. This section guides you through essential considerations and recommendations when adopting an IPAM tool.

Design considerations:

Numerous IPAM tools are available for your consideration, depending on your requirements and the size of your organization. The options spans from having a basic Excel-based inventory to open-source community-driven solution or comprehensive enterprise products with advanced features and support.

- Consider these factors when evaluating what IPAM tool to implement:
 - Minimum features required by your organization
 - Total cost of ownership (TCO), including licensing and ongoing maintenance
 - Audit trails, logging, and role-based access controls
 - Authentication and authorization through Microsoft Entra ID
 - Accessible via API
 - Integrations with other network management tools and systems
 - Active community support or the level of support from the software provider
- Consider evaluating an open-source IPAM tool like [Azure IPAM](#). Azure IPAM is a lightweight solution built on the Azure platform. It automatically discovers IP address utilization within your Azure tenant and enables you to manage it all from a centralized UI or via a RESTful API.
- Consider your organizations operating model and the ownership of the IPAM tool. The goal of implementing an IPAM tool is to streamline the process of requesting new IP address spaces for application teams without dependencies and bottlenecks.
- An important part of the IPAM tool functionality is to inventory IP address space usage and logically organize it.

Design recommendations:

- The process of reserving non-overlapping IP address spaces should support requesting different sizes based on the needs of the individual application landing zones.
 - For example, you could adopt T-shirt sizing to make it easy for application teams to describe their needs:
 - Small - /24 - 256 IP addresses
 - Medium - /22 - 1,024 IP addresses
 - Large - /20 - 4,096 IP addresses
- Your IPAM tool should have an API for reserving non-overlapping IP address spaces to support an Infrastructure as Code (IaC) approach. This feature is also crucial for seamless integration of IPAM into your [subscription vending process](#), thereby reducing the risk of errors and the need for manual intervention.
 - An example of an IaC approach is [Bicep](#) with its deployment script functionality or [Terraform](#) data sources to dynamically fetch data from the IPAM API.
- Create a systematic arrangement for your IP address spaces by structuring them according to Azure regions and workload archetypes, ensuring a clean and traceable network inventory.
- The decommissioning process for workloads should include the removal of IP address spaces that is no longer used, which can later be repurposed for upcoming new workloads, promoting efficient resource utilization.

Connectivity to Azure

Article • 06/28/2023

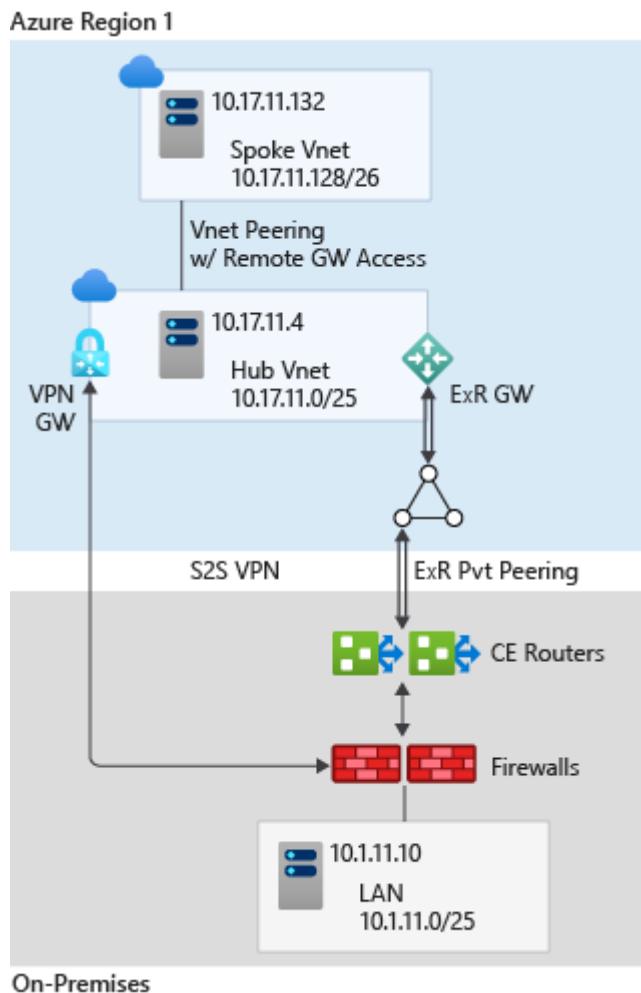
This section expands on the network topology to consider recommended models for connecting on-premises locations to Azure.

Design considerations:

- Azure [ExpressRoute](#) provides dedicated private connectivity to Azure infrastructure as a service (IaaS) and platform as a service (PaaS) functionality from on-premises locations.
- [Azure VPN \(S2S\) gateway](#) provides Site-to-Site shared connectivity over the public internet to Azure infrastructure as a service (IaaS) virtual networks from on-premises locations.
- [Azure ExpressRoute](#) and [Azure VPN \(S2S\)](#) have different capabilities, costs and performance, a [table](#) is available for comparison.
- You can use [private links](#) to establish connectivity to PaaS services, over ExpressRoute with private peering or VPN s2s from on-premises connected locations.
- When multiple virtual networks are connected to the same ExpressRoute circuit, they'll become part of the same [routing domain](#), and all virtual networks will share the bandwidth.
- You can use ExpressRoute [Global Reach](#), where available, to connect on-premises locations together through ExpressRoute circuits to transit traffic over the Microsoft backbone network.
- ExpressRoute [Global Reach](#) is available in many [ExpressRoute peering locations](#).
- [ExpressRoute Direct](#) allows creation of multiple ExpressRoute circuits at no additional cost, up to the ExpressRoute Direct port capacity (10 Gbps or 100 Gbps). It also allows you to connect directly to Microsoft's ExpressRoute routers. For the 100-Gbps SKU, the minimum circuit bandwidth is 5 Gbps. For the 10-Gbps SKU, the minimum circuit bandwidth is 1 Gbps.
- When enabled on an ExpressRoute circuit, [FastPath](#) sends network traffic directly to virtual machines in the virtual network, bypassing the gateway. FastPath is designed to improve the data path performance between your on-premises network and your virtual network without having a bottleneck on the gateway.

Design recommendations:

- Use ExpressRoute as the primary connectivity channel for connecting an on-premises network to Azure. You can use [VPNs as a source of backup connectivity](#) to enhance connectivity resiliency.



- Use dual ExpressRoute circuits from different peering locations when you're connecting an on-premises location to virtual networks in Azure. This setup will ensure redundant paths to Azure by removing single points of failure between on-premises and Azure.
- When you use multiple ExpressRoute circuits, [optimize ExpressRoute routing via BGP local preference and AS PATH prepending](#).
- Ensure that you're using the [right SKU](#) for the ExpressRoute/VPN gateways based on bandwidth and performance requirements.
- Deploy a [zone-redundant ExpressRoute gateway](#) in the supported Azure regions.
- For scenarios that require bandwidth higher than 10 Gbps or dedicated 10/100-Gbps ports, use [ExpressRoute Direct](#).

- When low latency is required, or throughput from on-premises to Azure must be greater than 10 Gbps, enable [FastPath](#) to bypass the ExpressRoute gateway from the data path.
- Use VPN gateways to connect branches or remote locations to Azure. For higher resilience, deploy [zone-redundant gateways](#) (where available).
- Use ExpressRoute [Global Reach](#) to connect large offices, regional headquarters, or datacenters connected to Azure via ExpressRoute.
- When traffic isolation or dedicated bandwidth is required, such as for separating production and nonproduction environments, use different ExpressRoute circuits. It will help you ensure isolated routing domains and alleviate noisy-neighbor risks.
- Use ExpressRoute [network insights](#) to monitor your ExpressRoute components (peerings, connections, gateways). ExpressRoute uses network insights to provide a detailed topology mapping of all ExpressRoute components (peerings, connections, gateways) and has preloaded metrics dashboard for availability, throughput, packet drops, and gateway metrics.
 - Use [Connection Monitor for ExpressRoute](#) to monitor connectivity between Azure cloud deployments and on-premises locations (branch offices, and so on.), detect network issues, identify and eliminate connectivity problems.
- Don't explicitly use ExpressRoute circuits from a single peering location. This creates a single point of failure and makes your organization susceptible to peering location outages.

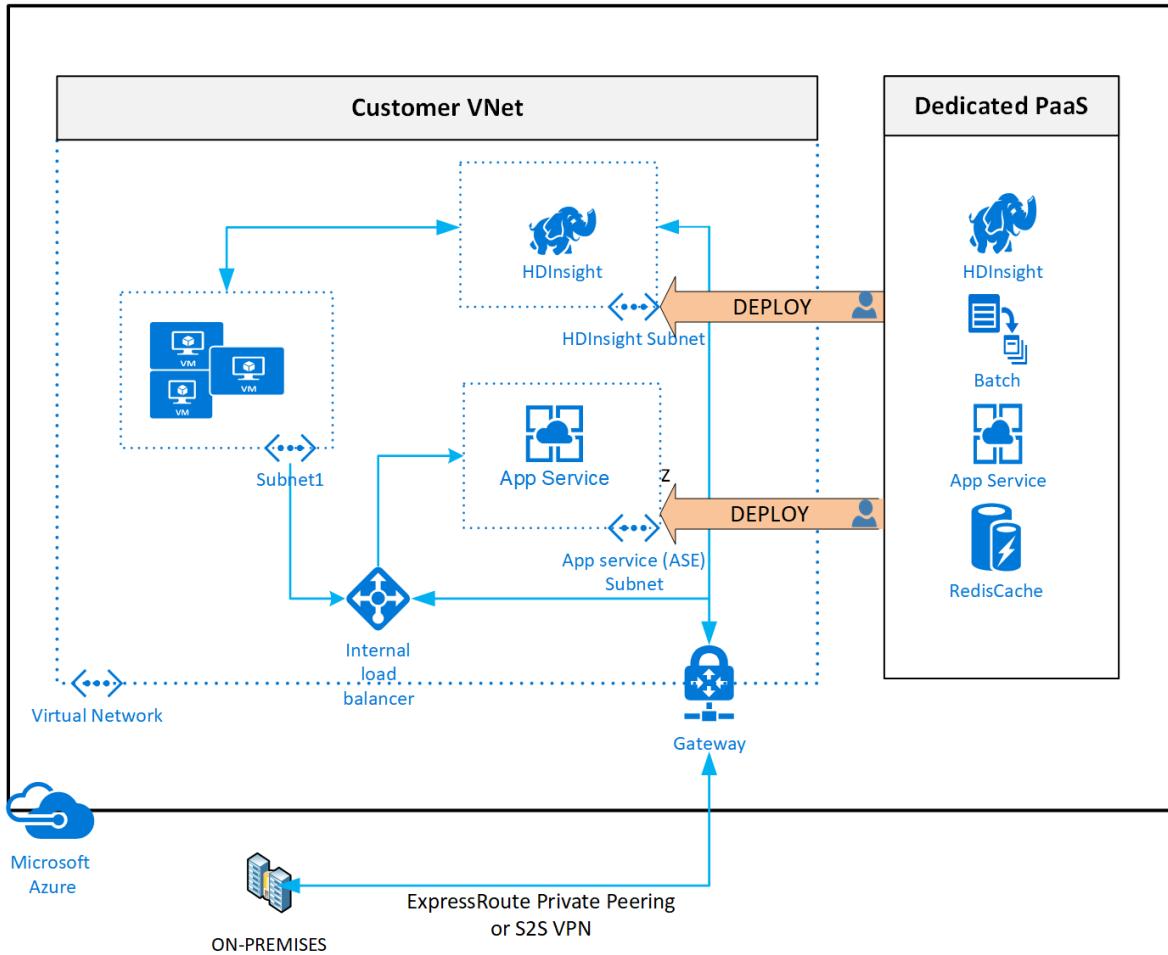
Connectivity to Azure PaaS services

Article • 05/25/2023

This article provides recommended connectivity approaches for using Azure PaaS services.

Design considerations

- Azure PaaS services are, in their default configuration, generally accessed over publicly available endpoints via the Microsoft Global Network. Some customers may have requirements to reduce the usage of public endpoints, therefore the Azure platform provides optional capabilities for securing these endpoints or even making them entirely private.
 - Some PaaS services allow public access restrictions based on **Resource Instance** system-assigned managed identity E.g. [Azure Storage](#)
 - Many PaaS services allow public access restrictions based on **Trusted Azure Services** E.g. [Azure Container Registry](#)
 - **Virtual network injection** provides [dedicated private deployments](#) for supported services. Management plane traffic still flows through public IP addresses.



- Some PaaS services are compatible with [Azure Private Link](#) which allows private access via an IP address within a customer. For more information, see [Key benefits of Private Link](#).
- [Virtual network service endpoints](#) provide service-level access from selected subnets to selected PaaS services. Azure Storage offers [Service Endpoint Policies](#) which allow further restricting the use of Service Endpoints to specific Storage Account. It is also possible to utilize Network Virtual Appliances (NVA) to perform Layer-7 inspection and FQDN filtering in combination with Service Endpoints, but this approach comes with additional performance and scaling considerations.

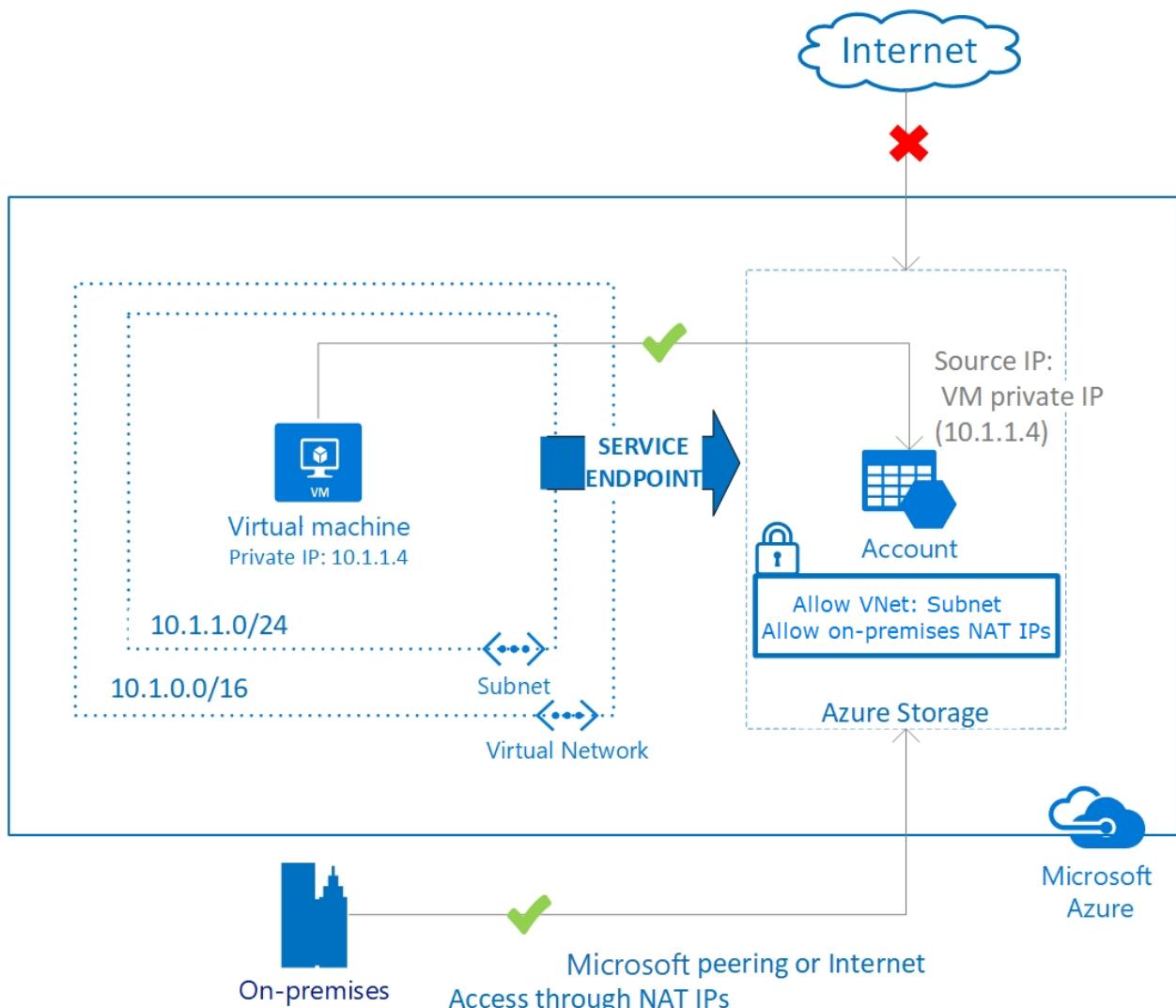
[What is the difference between service endpoints and private endpoints?](#) offers an explanation of differences between Private Link endpoints and VNet service endpoints.

Design recommendations

- For Azure PaaS services that support virtual network injection, if you require access to resources within your private network (either virtual networks or on-premises via a virtual network gateway), consider enabling the virtual network injection

feature. Also consider that these services injected into a virtual network still perform management plane operations by using service specific public IP addresses. Connectivity must be guaranteed for the service to operate correctly. Use UDRs and NSGs to lock down this communication within the virtual network. You can use [Service Tags in UDR](#) to reduce the number of necessary routes and to override default routes if used.

- When data exfiltration protection and use of only Private IP addressing are firm requirements, consider the use of Azure Private Link [where available](#).
- Consider the use of virtual network service endpoints to secure access to Azure PaaS services from within your virtual network in scenarios where data exfiltration is less of a concern, Private Link is unavailable, or you have a requirement for large data ingest that requires cost optimization. (Azure Service Endpoints do not incur any costs, contrasted to Azure Private Link which includes a cost component based on per GB of network data).



- If access to Azure PaaS services is required from on-premises utilize the following options:

- Use the PaaS service's default public endpoint via the Internet and the Microsoft Global Network if no private access is required and the on-premises Internet bandwidth is sufficient.
- Use a private hybrid connection ([ExpressRoute with private peering](#) or Site-to-Site VPN) with either virtual network injection or Azure Private Link.
- Don't enable virtual network service endpoints by default on all subnets. Follow the above considered approach on a case-by-case basis dependent on the PaaS service feature availability and your own performance and security requirements.
- Where possible, avoid the use of forced tunneling (directing Internet-bound traffic from an Azure virtual network via on-premises by advertising a default route over a private hybrid connection) as this can increase the complexity of managing control-plane operations with some Azure PaaS services E.g. [Application Gateway V2](#).

Limit cross-tenant private endpoint connections in Azure

Article • 10/09/2023

Customers are increasingly using private endpoints in their tenants to connect to their Azure platform as a service (PaaS) services privately and securely. Private endpoints can connect to services across Microsoft Entra tenants. For security and compliance, you might need to block cross Microsoft Entra tenants connections on your private endpoints. This guidance shows you recommended configuration options to limit or prevent cross-tenant private endpoint connections. These options help you create data leakage prevention (DLP) controls inside your Azure environment.

Introduction to private endpoints

Use private endpoints to control the traffic within your Azure environment using an existing network perimeter. But there are scenarios where you must keep private endpoint connections within the corporate Microsoft Entra tenant only. The following examples show connections that might create security risks.

- **Connection A:** A rogue administrator creates private endpoints on the customer virtual network. These endpoints link to services that are hosted outside the customer environment, like another Microsoft Entra tenant.
- **Connection B:** A rogue administrator creates private endpoints in other Microsoft Entra tenants that link to services hosted in the customer's Microsoft Entra tenant.

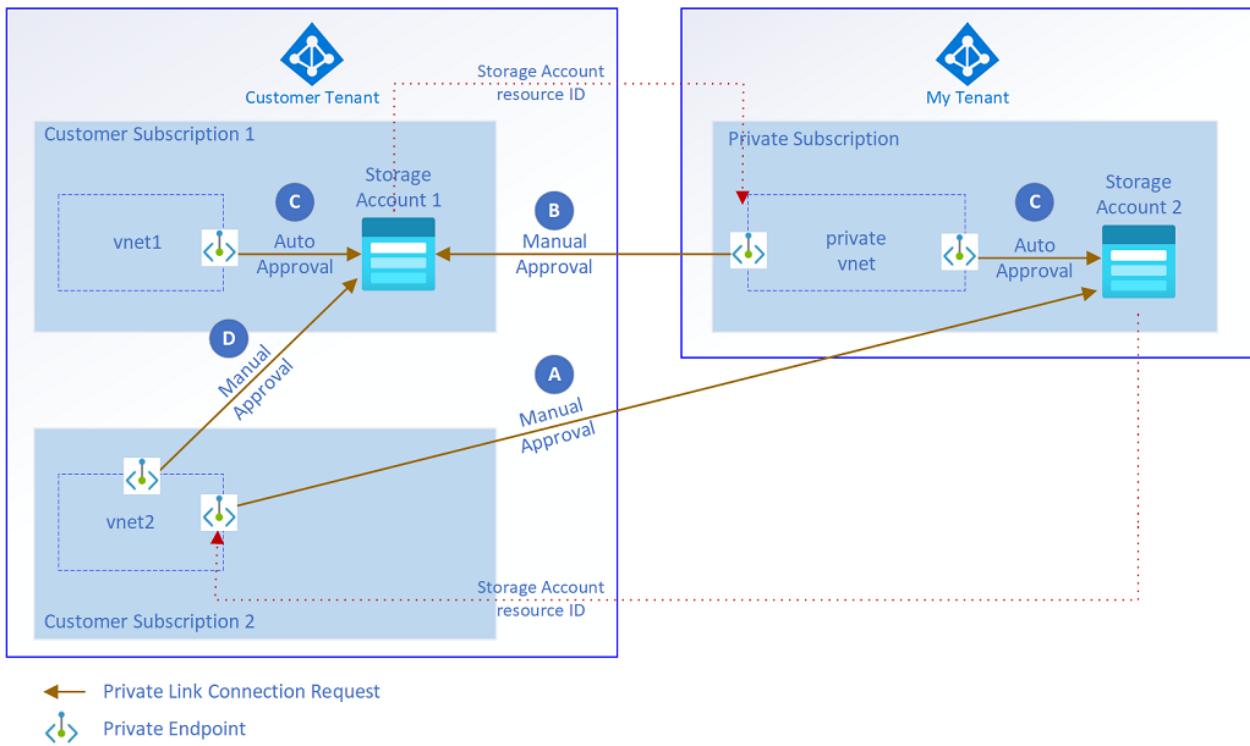


Figure 1: Illustration of private endpoint cross-tenant scenarios.

For both scenarios, you specify the resource ID of the service and manually approve the private endpoint connection. Users also require role-based access control (RBAC) access to run these actions.

Connections C and D in Figure 1 show scenarios that customers generally want to allow. The private endpoint connections are kept within the corporate Microsoft Entra tenant. They don't represent a security risk so these two scenarios aren't covered in this article.

The following information gives you options to prevent private endpoints provisioning across Microsoft Entra tenants.

Deny private endpoints linked to services in other tenants

Scenario one: A rogue administrator requires the following rights in a subscription in the customer's Microsoft Entra tenant.

- **Microsoft.Network/virtualNetworks/join/action** rights on a subnet with **privateEndpointNetworkPolicies** set to **Disabled**.
- **Microsoft.Network/privateEndpoints/write** access to a resource group in the customer environment.

With these rights, a rogue administrator can create a private endpoint in the customer's Microsoft Entra tenant. This private endpoint links to a service in a separate subscription

and Microsoft Entra tenant. Figure 1 shows this scenario as connection A.

For this scenario, the user sets up an external Microsoft Entra tenant and Azure subscription. Next, they create a private endpoint in the customer environment by manually specifying the resource ID of the service. Finally, the rogue administrator approves the private endpoint on the linked service that's hosted in the external Microsoft Entra tenant to allow traffic over the connection.

After the rogue administrator approves the private endpoint connection, corporate data can be copied from the corporate virtual network to an Azure service on an external Microsoft Entra tenant. This security risk can only occur if access was granted using Azure RBAC.

Mitigation for scenario one

Use the following [Azure Policy](#) to automatically block the ability to create a private endpoint in the corporate Microsoft Entra tenant that's linked to an outside Azure service.

JSON

```
"if": {
  "allof": [
    {
      "field": "type",
      "equals": "Microsoft.Network/privateEndpoints"
    },
    {
      "anyOf": [
        {
          "count": {
            "field":
"Microsoft.Network/privateEndpoints/manualprivateLinkServiceConnections[*]",
            "where": {
              "allof": [
                {
                  "field":
"Microsoft.Network/privateEndpoints/manualprivateLinkServiceConnections[*].p
rivateLinkId",
                  "notEquals": ""
                },
                {
                  "value": "
[split(concat(first(field('Microsoft.Network/privateEndpoints/manualprivateL
inkServiceConnections[*].privateLinkId')),'//'), '/')][2]]",
                  "notEquals": "
[subscription().subscriptionId]"
                }
              ]
            }
          }
        }
      ]
    }
  ]
}
```

```

        }
    },
    "greaterOrEquals": 1
},
{
    "count": {
        "field":
"Microsoft.Network/privateEndpoints/privateLinkServiceConnections[*]",
        "where": {
            "allOf": [
                {
                    "field":
"Microsoft.Network/privateEndpoints/privateLinkServiceConnections[*].private
LinkServiceId",
                    "notEquals": ""
                },
                {
                    "value": "
[split(concat(first(field('Microsoft.Network/privateEndpoints/privateLinkSer
viceConnections[*].privateLinkId')), '//'), '/')][2]]",
                    "notEquals": "
[subscription().subscriptionId]"
                }
            ]
        }
    },
    "greaterOrEquals": 1
}
]
}
],
{
    "then": {
        "effect": "Deny"
}

```

This policy denies any private endpoints created outside of the subscription of the linked service, like connections A and D. The policy also provides the flexibility to use `manualPrivateLinkServiceConnections` and `privateLinkServiceConnections`.

You can update this policy so private endpoints are only created in a certain set of subscriptions. You can make this change by adding a `list` parameter and use the `"notIn": "[parameters('allowedSubscriptions')]"` construct. But this approach isn't recommended, because it means that you'd have to constantly maintain the list of subscriptions for this policy. Whenever a new subscription is created inside your tenant, the subscription ID must be added to the parameter.

Instead, assign the policy to the top-level management group, and then use exemptions where required.

Considerations for scenario one

This policy blocks the ability to create private endpoints that are in a different subscription than the service itself. If these endpoints are required for certain use cases, use policy exemptions. Create more policies for Data Factory and Azure Synapse to make sure that managed private endpoints hosted on the managed virtual network can only connect to services hosted within your Microsoft Entra tenant.

Deny connections from private endpoints created in other tenants

Scenario two: A rogue administrator requires **write** access on the service in the customer environment for which a private endpoint should be created.

With this right, a rogue administrator can create a private endpoint in an external Microsoft Entra tenant and subscription. This endpoint links to a service in the customer's Microsoft Entra tenant. Figure 1 shows this scenario as connection B.

In this scenario, the rogue administrator needs to first configure an external private Microsoft Entra tenant and Azure subscription. Next, they create a private endpoint in their environment by manually specifying the resource ID and group ID of the service in the corporate Microsoft Entra tenant. Finally, they approve the private endpoint on the linked service to allow traffic over the connection across Microsoft Entra tenants.

After the rogue administrator or service owner approves the private endpoint, data is accessed from the external virtual network.

Mitigation for scenario two

Use service-specific policies to prevent this scenario across the customer tenant. Private endpoint connections are subresources of the respective services and show up under their properties section. Deny noncompliant connections by using the following [policy definition](#):

JSON

```
"if": {
  "allof": [
    {
      "field": "type",
      "equals":
        "Microsoft.Storage/storageAccounts/privateEndpointConnections"
    },
  ]
}
```

```

{
    "field": "Microsoft.Storage/storageAccounts/privateEndpointConnections/privateLinkServiceConnectionState.status",
        "equals": "Approved"
},
{
    "anyOf": [
        {
            "field": "Microsoft.Storage/storageAccounts/privateEndpointConnections/privateEndpoint.id",
                "exists": false
            },
            {
                "value": "[split(concat(field('Microsoft.Storage/storageAccounts/privateEndpointConnections/privateEndpoint.id'), '//'), '/')][2]]",
                    "notEquals": "[subscription().subscriptionId]"
                }
            ]
        }
    ],
    "then": {
        "effect": "Deny"
}

```

This policy shows an example for Azure Storage. Replicate the same policy definition for other services like [Key Vault](#), [cognitive services](#), and [SQL Server](#). Note that Azure App Service doesn't support this mitigation at this time.

To further improve manageability, bundle the service-specific policies into an initiative. The policy denies the approval of private endpoint connections to private endpoints that are hosted outside of the subscription of the respective service. It doesn't deny the rejection or removal of private endpoint connections, which is the behavior customers want. Auto-approval workflows, such as connection C, aren't affected by this policy.

But the approval of compliant private endpoint connections within the portal is blocked with this method. This block occurs because the portal UI doesn't send the resource ID of the connected private endpoint in their payload. It's recommended to use [Azure Resource Manager](#), [Azure PowerShell](#), or [Azure CLI](#) to approve the private endpoint connection.

Also, assign the policy to the top-level management group and use exemptions where required.

Considerations for scenario two

Azure Synapse Analytics and Azure Data Factory offer managed virtual networks and managed private endpoints. Because of these new capabilities, the policy blocks the secure and private usage of these services.

It's recommended that you use an **Audit** effect instead of a **Deny** effect in the policy definition you use in the [scenario two mitigation](#). This change helps you keep track of private endpoints being created in separate subscriptions and tenants. You can also use policy exemptions for the respective data platform scopes.

Azure Data Factory

To overcome [scenario one](#) on the managed virtual network of Azure Data Factory, use the following [policy definition](#):

JSON

```
"if": {
  "allof": [
    {
      "field": "type",
      "equals":
        "Microsoft.DataFactory/factories/managedVirtualNetworks/managedPrivateEndpoints"
    },
    {
      "anyof": [
        {
          "field":
            "Microsoft.DataFactory/factories/managedVirtualNetworks/managedPrivateEndpoints/privateLinkId",
          "exists": false
        },
        {
          "value": "
[split(field('Microsoft.DataFactory/factories/managedVirtualNetworks/managedPrivateEndpoints/privateLinkId'), '/')][2]]",
          "notequals": "[subscription().subscriptionId]"
        }
      ]
    }
  ],
  "then": {
    "effect": "[parameters('effect')]"
  }
}
```

This policy denies managed private endpoints that are linked to services, which are hosted outside the subscription of the Data Factory. You can change this policy to allow

connections to services hosted in a set of subscriptions by adding a `list` parameter and by using the `"notIn": "[parameters('allowedSubscriptions')]"` construct. We recommend this change for the data platform scope inside the tenant or environments where services with managed virtual networks and managed private endpoints are extensively used.

It's recommended that you assign this policy to the top-level management group and use exemptions where required. For the data platform, make these changes and assign the policy to the set of data platform subscriptions.

Azure Synapse

Azure Synapse also uses managed virtual networks. We recommend applying a similar policy to the Data Factory policy for [scenario one](#). Azure Synapse doesn't provide a policy alias for managed private endpoints. But there's a data exfiltration prevention feature, which can be enforced for workspaces using the following policy:

JSON

```
"if": {
  "allOf": [
    {
      "field": "type",
      "equals": "Microsoft.Synapse/workspaces"
    },
    {
      "anyOf": [
        {
          "field":
"Microsoft.Synapse/workspaces/managedVirtualNetworkSettings.preventDataExfil
tration",
          "exists": false
        },
        {
          "field":
"Microsoft.Synapse/workspaces/managedVirtualNetworkSettings.preventDataExfil
tration",
          "notEquals": true
        }
      ]
    },
    {
      "count": {
        "field":
"Microsoft.Synapse/workspaces/managedVirtualNetworkSettings.allowedAadTenant
IdsForLinking[*]",
        "where": {
          "field":
"Microsoft.Synapse/workspaces/managedVirtualNetworkSettings.allowedAadTenant
IdsForLinking[*]",
          "notEquals": "[subscription().tenantId]"
        }
      }
    }
  ]
}
```

```
        }
      ],
    }
  ],
},
"then": {
  "effect": "Deny"
}
}
```

This policy enforces the use of the data exfiltration feature of Azure Synapse. With Azure Synapse, you can deny any private endpoint that's coming from a service that's hosted outside of the customer tenant. You can also deny any private endpoint hosted outside of a specified set of tenant IDs. This policy only allows creating managed private endpoints that are linked to services, which are hosted in the customer tenant.

These policies are now available as built-in.

- Azure Synapse workspaces should allow outbound data traffic only to approved targets.

Definition ID: `/providers/Microsoft.Authorization/policyDefinitions/3484ce98-c0c5-4c83-994b-c5ac24785218`

- Azure Synapse managed private endpoints should only connect to resources in approved Microsoft Entra tenants.

Definition ID: `/providers/Microsoft.Authorization/policyDefinitions/3a003702-13d2-4679-941b-937e58c443f0`

It's recommended that you assign the policy to the top-level management group and use exemptions where required.

Next steps

It's important to understand the recommended connectivity models for inbound and outbound connectivity to and from the public internet. The next article reviews design considerations, design recommendations, and recommended content for further reading.

[Inbound and outbound connectivity](#)

Connectivity to other cloud providers

Article • 12/01/2022

This guidance discusses ways to connect an Azure landing zone architecture to other cloud providers, such as Amazon Web Services (AWS) and Google Cloud Platform (GCP).

The various options differ in speed, latency, reliability, service-level agreements (SLAs), complexity, and costs. This article considers options and makes recommendations.

ⓘ Note

Microsoft and Oracle partnered to provide high-throughput, low-latency cross-connections between Azure and Oracle Cloud Infrastructure (OCI). For more information, see [Connectivity to Oracle Cloud Infrastructure](#).

Design considerations

- We consider the following options to connect Azure to another cloud:
 - **Option 1:** Connect Azure ExpressRoute and the other cloud provider's equivalent private connection. The customer manages routing.
 - **Option 2:** Connect ExpressRoute and the other cloud provider's equivalent private connection. A cloud exchange provider handles routing.
 - **Option 3:** Use Site-to-Site VPN over the internet. For more information, see [Connect on-premises networks to Azure by using Site-to-Site VPN gateways \(Learn\)](#).

You can use the following cross-cloud connectivity flow chart as an aid to choosing an option:

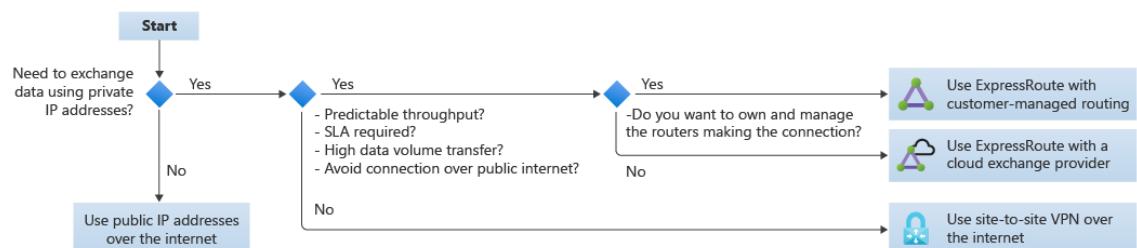


Figure 1: Cross-cloud connectivity flow chart

- You can only connect an Azure virtual network to another cloud provider's virtual private cloud (VPC) if the private IP address spaces don't overlap.
- Site-to-Site VPN might have lower throughput and higher latency than the ExpressRoute options.
- Site-to-Site VPN is the fastest deployment option if Azure ExpressRoute and the other cloud provider equivalent aren't already in use.
- Routing complexity of Azure ExpressRoute and other cloud provider equivalent with customer-managed routing can be high if not done through a cloud exchange provider.
- All options are applicable to both the traditional Azure network topology and the Virtual WAN topology.
- You might need to provide DNS resolution between Azure and the other cloud provider. This configuration might incur extra costs.
- The FastPath feature of ExpressRoute improves data path performance between Azure and on-premises networks, and between Azure and other cloud providers. When enabled, FastPath sends network traffic directly to virtual machines in the virtual network, bypassing the ExpressRoute gateway. For more information, see [About ExpressRoute FastPath](#).
- FastPath is available on all ExpressRoute circuits.
- FastPath still requires a virtual network gateway to be created for route exchange purposes. The virtual network gateway must use either the Ultra Performance SKU or the ErGw3AZ SKU for the ExpressRoute gateway to enable route management.
- There are configurations that FastPath doesn't support, such as a UDR on the gateway subnet. For more information, see [Limitations](#) in about ExpressRoute FastPath.

Design recommendations

- Use option 1 or option 2 to avoid use of the public internet, if you require an SLA, if you want predictable throughput, or need to handle data volume transfer. Consider whether to use a customer-managed routing or a cloud exchange provider if you haven't implemented ExpressRoute already.
- Create the ExpressRoute circuits for option 1 and option 2 in the connectivity subscription.

- Use the ExpressRoute circuit of option 1 or option 2 to connect to the hub virtual network of a traditional hub and spoke topology or to the virtual hub for a Virtual WAN topology. For more information, see Figure 2 and Figure 3.

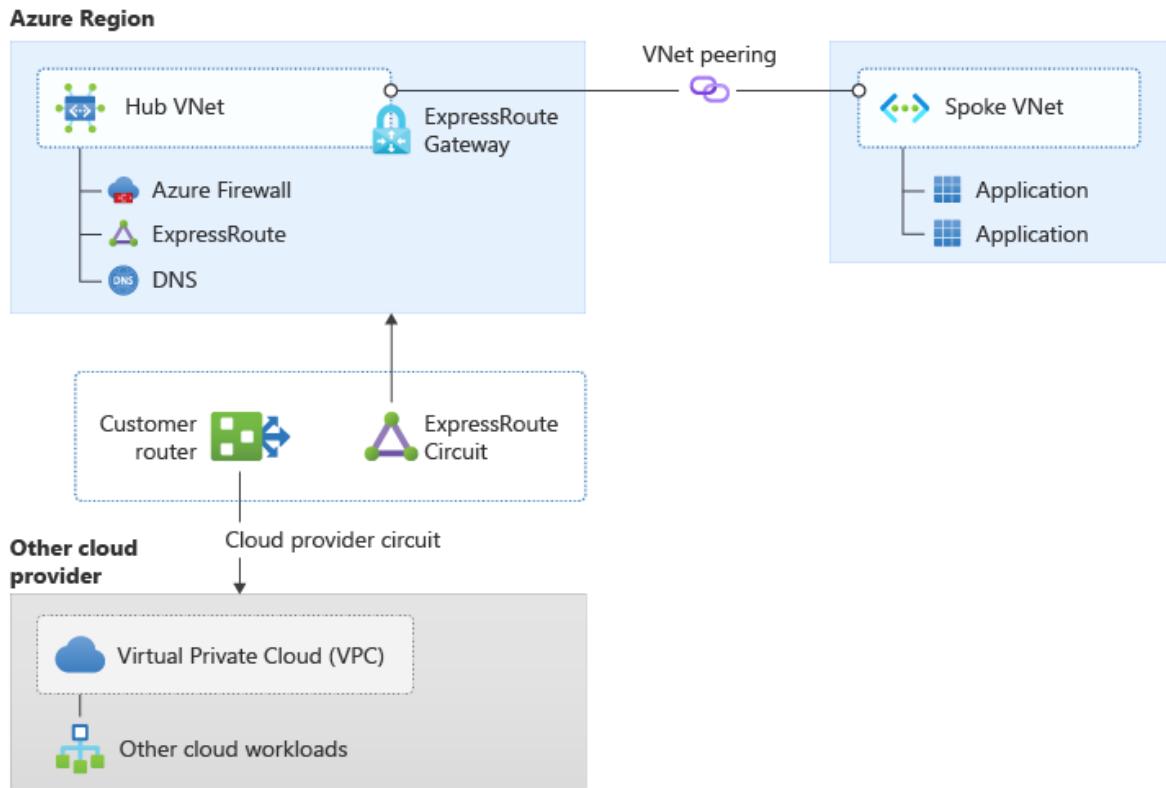


Figure 2: Cross-cloud connectivity with customer-managed routing (Option 1)

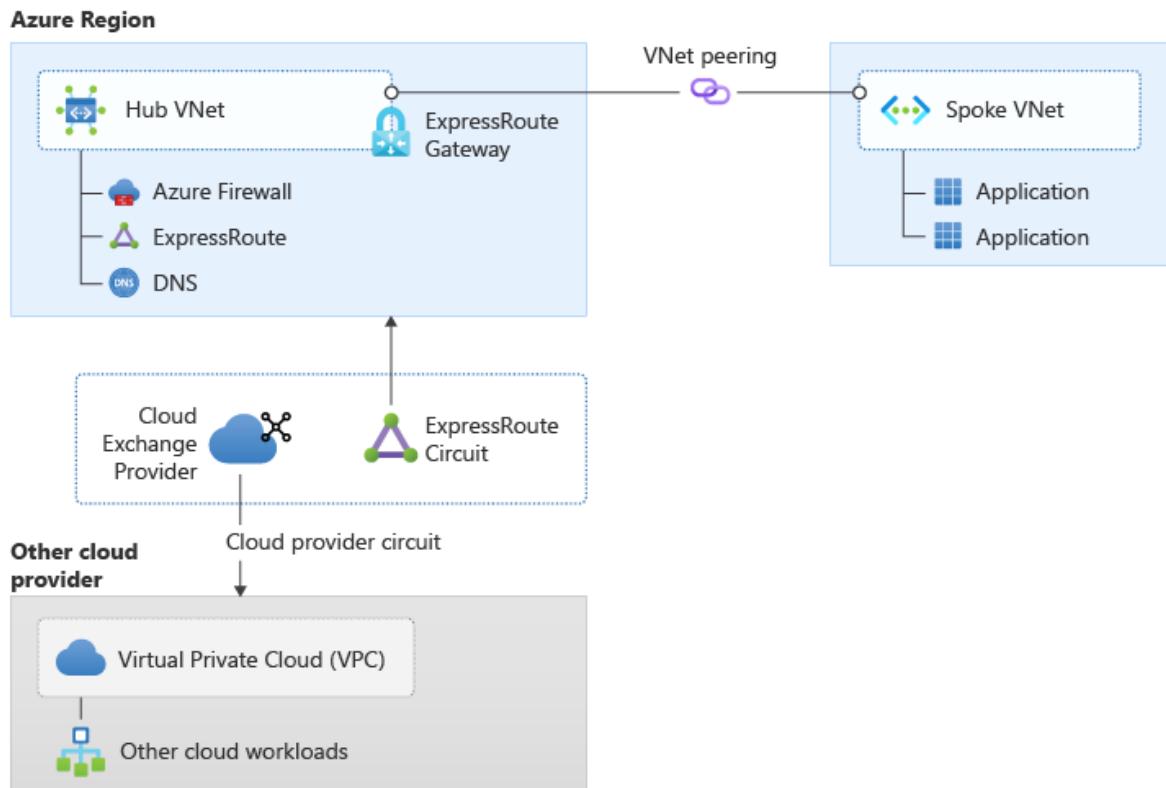


Figure 3: Cross-cloud connectivity with a cloud exchange provider (Option 2)

- If you need to minimize latency between Azure and another cloud provider, consider deploying your application in a single virtual network with an ExpressRoute gateway, and enable FastPath.

Azure Region

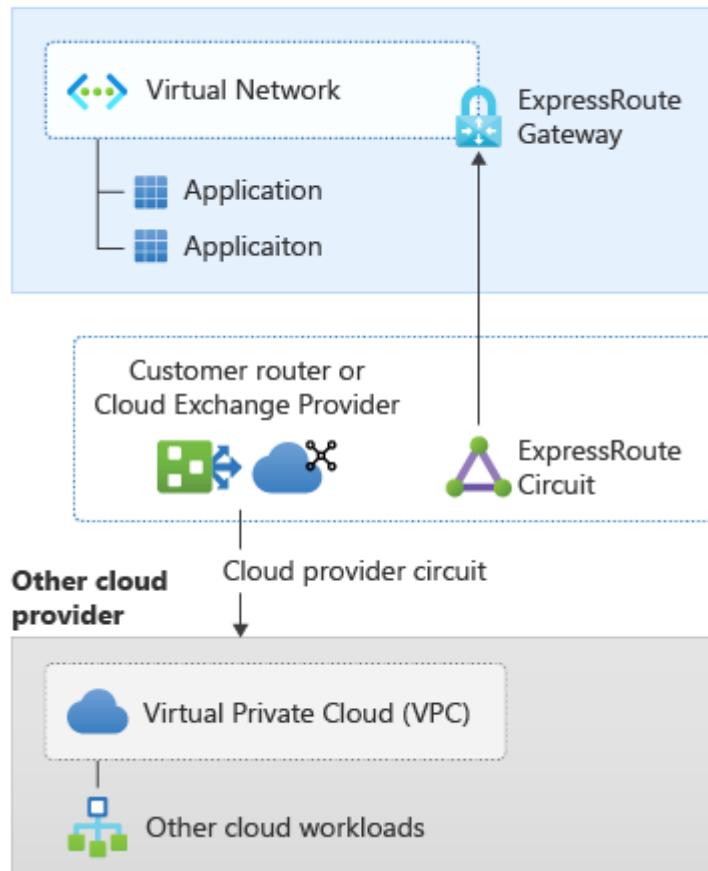


Figure 4: Cross-cloud connectivity with FastPath enabled

- If ExpressRoute isn't required or not available, you can use Site-to-Site VPN over the internet to connect between Azure and another cloud provider.

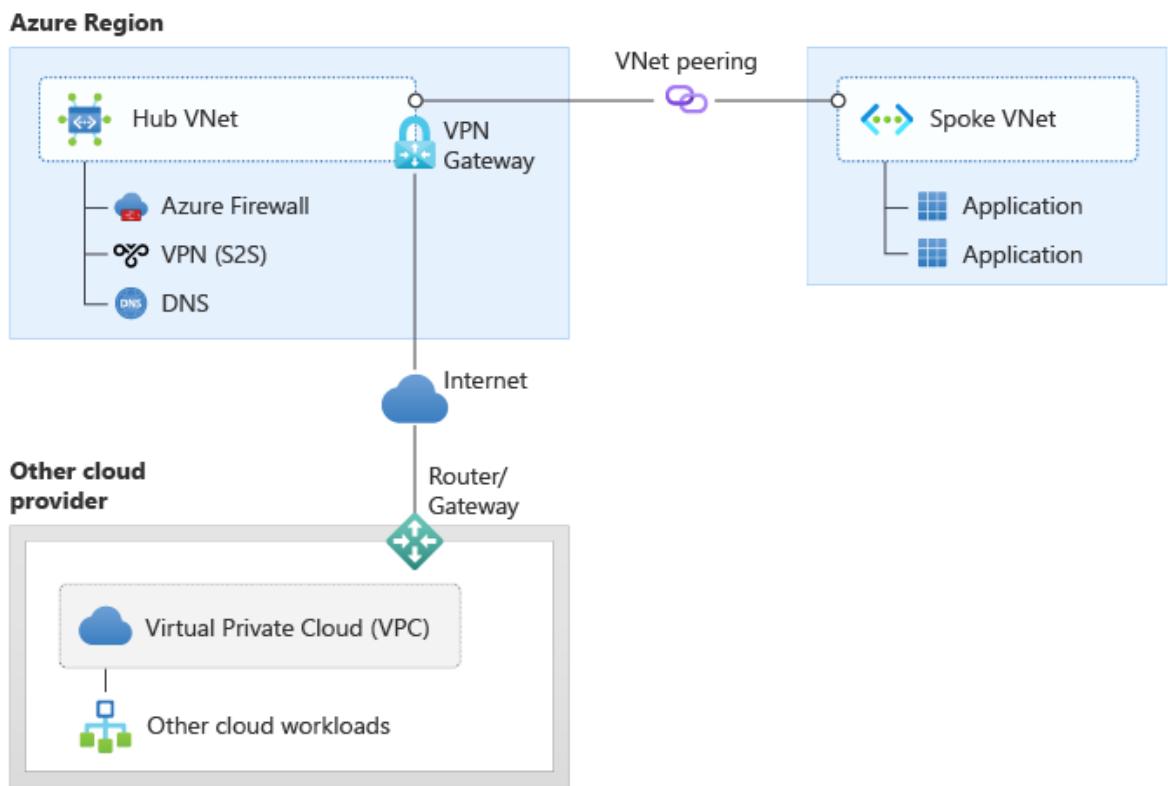


Figure 5: Cross-cloud connectivity using site-to-site VPN over the Internet

Next steps

To learn more about connectivity to Oracle Cloud Infrastructure (OCI), see [Connectivity to Oracle Cloud Infrastructure](#).

Connectivity to Oracle Cloud Infrastructure

Article • 12/01/2022

This section provides different connectivity approaches to integrate an Azure landing zone architecture to Oracle Cloud Infrastructure (OCI).

Design considerations:

- Using ExpressRoute and FastConnect, customers can connect a virtual network in Azure with a virtual cloud network in OCI, if the private IP address space doesn't overlap. Once you establish connectivity, resources in the Azure virtual network can communicate with resources in the OCI virtual cloud network as if they were both in the same network.
- Azure ExpressRoute [FastPath](#) is designed to improve the data path performance between two networks, both on-premises and Azure, and for this scenario, between OCI and Azure. When enabled, FastPath sends network traffic directly to virtual machines in the virtual network, bypassing the ExpressRoute gateway.
 - FastPath is available on all ExpressRoute circuits.
 - FastPath still requires a virtual network gateway to be created for route exchange purposes. The virtual network gateway must use either the Ultra Performance SKU or the ErGw3AZ SKU for the ExpressRoute gateway to enable route management.
- There are features that are currently [not supported](#) in ExpressRoute FastPath, such as Azure Virtual WAN hubs or VNet peering.
- While you can use [ExpressRoute Global Reach](#) to enable communication from on-premises to OCI via ExpressRoute circuits, it might incur more bandwidth costs that you can calculate by using the [Azure pricing calculator](#). It's important to consider any extra costs when you migrate large amounts of data from on-premises to Oracle by using ExpressRoute circuits.
- In Azure regions that support [Availability Zones](#), placing your Azure workloads in one zone or the other can have a small effect on latency. Design your application to balance availability and performances requirements.
- Interconnectivity between Azure and OCI is only available for [specific regions](#).

- For more in-depth documentation about interconnectivity between Azure and OCI, see [Oracle application solutions to integrate Microsoft Azure and Oracle Cloud Infrastructure](#) or see [Access to Microsoft Azure in OCI](#).

Design recommendations:

- Create the ExpressRoute circuit that will be used to interconnect Azure with OCI in the **connectivity** subscription.
- You can interconnect an Azure network architecture based on the traditional hub and spoke architecture or Azure Virtual WAN-based network topologies. It can be done by connecting the ExpressRoute circuit that will be used to interconnect Azure to OCI to the hub VNet or Virtual WAN hub as shown in the following diagram.

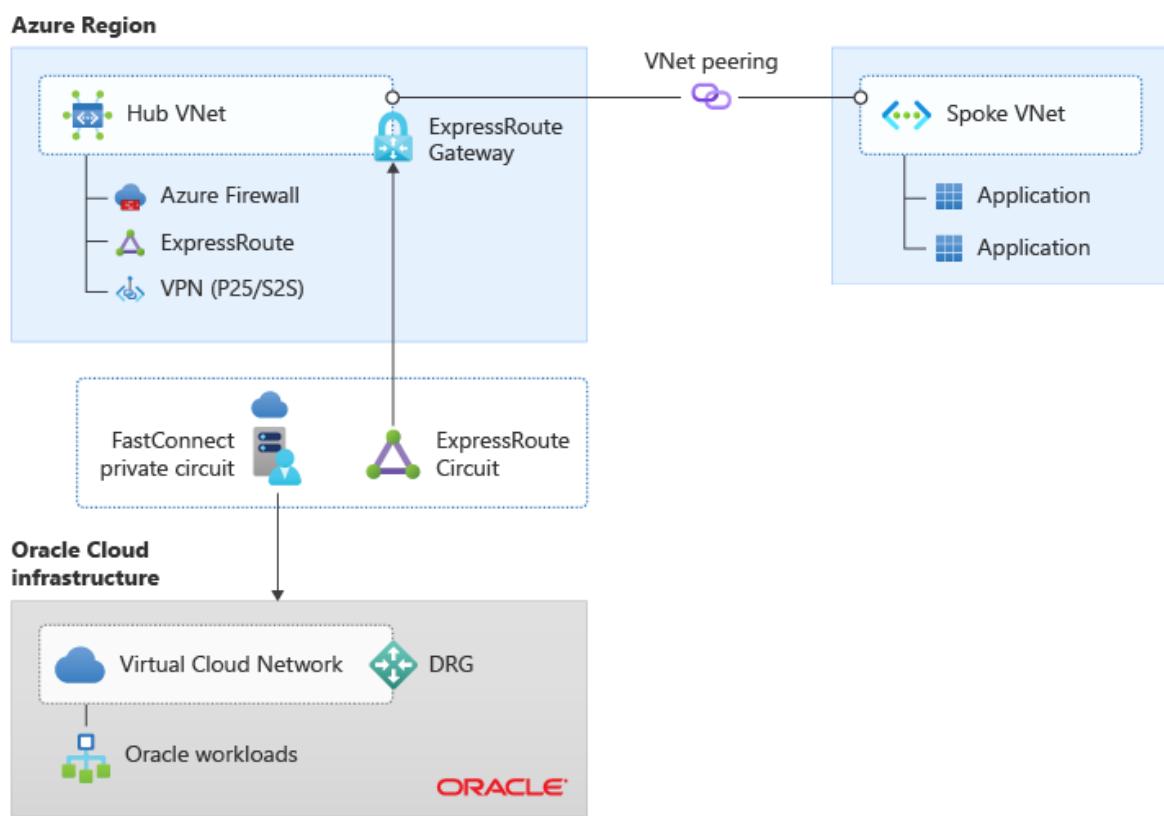


Figure 1: Interconnectivity between Azure and OCI via ExpressRoute.

- If your application requires the lowest possible latency between Azure and OCI, consider deploying your application in a single VNet with an ExpressRoute gateway and FastPath enabled.

Azure Region

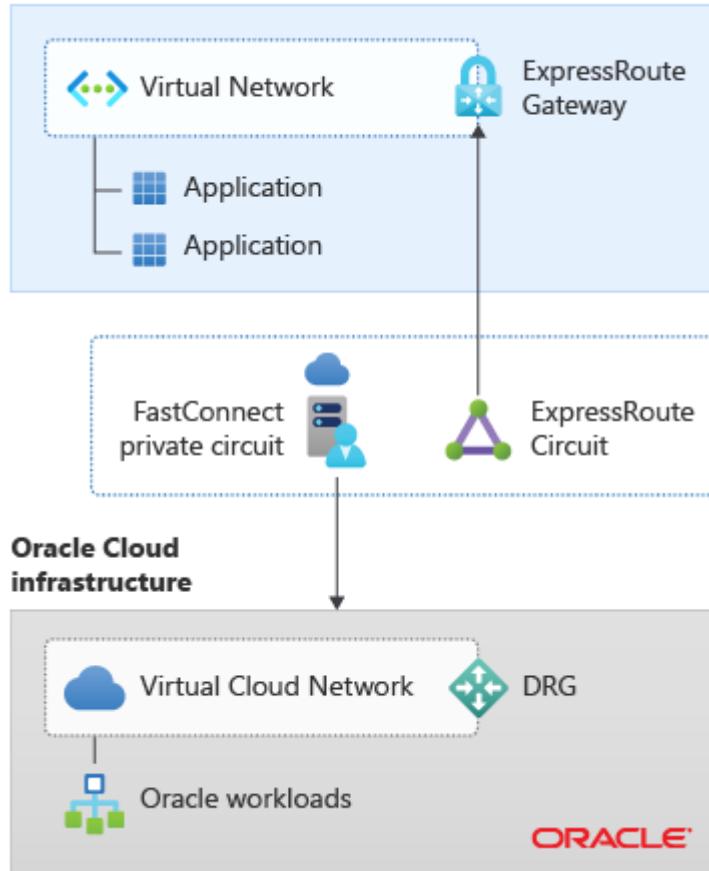


Figure 2: Interconnectivity between Azure and OCI with a single VNet.

- When you deploy Azure resources across Availability Zones, perform latency tests from Azure VMs located in different Availability Zones to OCI resources to understand which of the three Availability Zones provides the lowest latency to the OCI resources.
- To operate Oracle resources hosted in OCI by using Azure resources and technologies, you could:
 - **From Azure:** Deploy a jump box in a spoke VNet. The jump box provides access to the virtual cloud network in OCI as shown in the following picture:

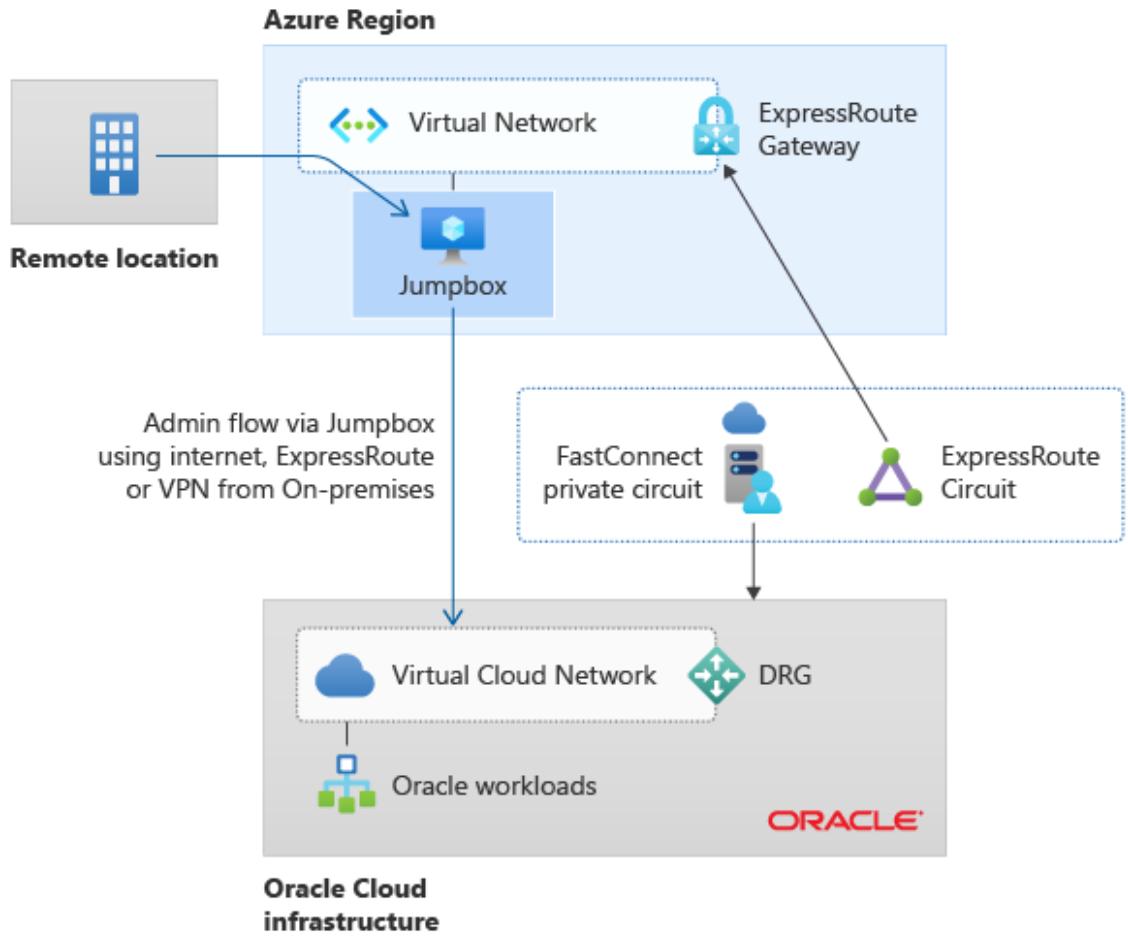


Figure 3: Managing OCI resources from Azure via a jump box.

- **From on-premises:** Use ExpressRoute Global Reach to bind an existing ExpressRoute circuit that connects on-premises to Azure, to an OCI ExpressRoute circuit that interconnects Azure to OCI. In this way, the Microsoft Enterprise Edge (MSEE) router becomes the central routing point between both ExpressRoute circuits.

Azure Region

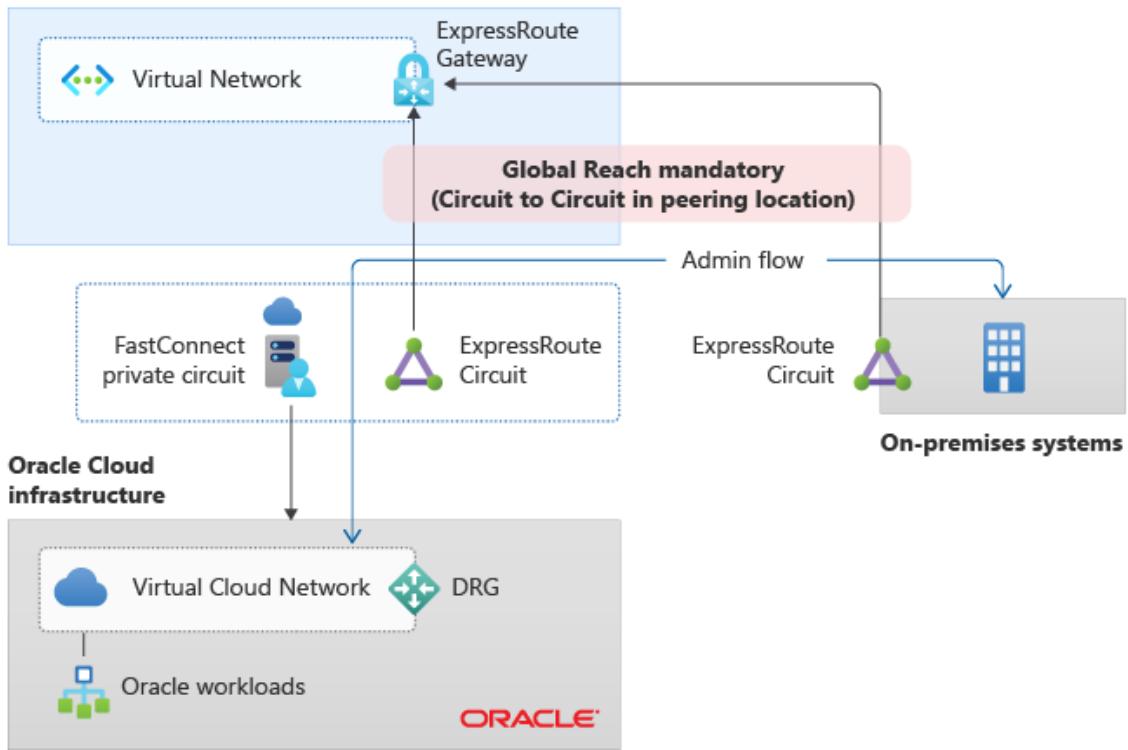


Figure 4: Managing OCI resources from on-premises via ExpressRoute Global Reach.

Next steps

For information on connectivity to other cloud providers, see [Connectivity to other cloud providers](#).

Plan for application delivery

Article • 03/13/2023

This section explores key recommendations to deliver internal-facing and external-facing applications in a secure, highly scalable, and highly available way.

Design considerations:

- Azure Load Balancer (internal and public) provides high availability for application delivery at a regional level.
- Azure Application Gateway allows the secure delivery of HTTP/S applications at a regional level.
- Azure Front Door allows the secure delivery of highly available HTTP/S applications across Azure regions.
- Azure Traffic Manager allows the delivery of global applications.

Design recommendations:

- Perform application delivery within landing zones for both internal-facing and external-facing applications.
 - Treat the Application Gateway as an application component and deploy it in a spoke virtual network not as a shared resource in the hub.
 - To interpret Web Application Firewall alerts, you generally need in-depth knowledge of the application to decide whether the messages that trigger those alerts are legitimate.
 - You might face role-based access control problems if you deploy Application Gateway in the hub when teams manage different applications but use the same instance of Application Gateway. Each team then has access to the entire Application Gateway configuration.
 - If you treat Application Gateway as a shared resource, you might exceed [Azure Application Gateway limits](#).
 - Read more about this in [Zero-trust network for web applications](#).
- For secure delivery of HTTP/S applications, use Application Gateway v2 and ensure that WAF protection and policies are enabled.
- Use a partner NVA if you can't use Application Gateway v2 for the security of HTTP/S applications.

- Deploy Azure Application Gateway v2 or partner NVAs used for inbound HTTP/S connections within the landing-zone virtual network and with the applications that they're securing.
- Use a DDoS standard protection plan for all public IP addresses in a landing zone.
- Use Azure Front Door with WAF policies to deliver and help protect global HTTP/S applications that span Azure regions.
- When you're using Front Door and Application Gateway to help protect HTTP/S applications, use WAF policies in Front Door. Lock down Application Gateway to receive traffic only from Front Door.
- Use Traffic Manager to deliver global applications that span protocols other than HTTP/S.

Plan for inbound and outbound internet connectivity

Article • 04/21/2023

This article lists considerations and recommendations for inbound and outbound connectivity between Azure and the public internet.

Design considerations

- Azure native network security services such as [Azure Firewall](#), [Azure Web Application Firewall \(WAF\)](#) on [Azure Application Gateway](#), and [Azure Front Door](#) are fully managed. You don't incur the operational and management costs and complexity of infrastructure deployments at scale.
- If your organization prefers to use non-Azure network virtual appliance (NVAs), or for situations where native services don't satisfy specific requirements, the Azure landing zone architecture is fully compatible with partner NVAs.
- Azure provides several direct internet outbound connectivity methods, such as network address translation (NAT) gateways or load balancers, for virtual machines (VMs) or compute instances on a virtual network. [Azure NAT Gateway](#) is recommended as the default for enabling outbound connectivity as it is operationally the simplest to set up, and is the most scalable and efficient option among all outbound connectivity methods available in Azure. For more information, see [Azure outbound connectivity methods](#).

Design recommendations

- Use Azure NAT Gateway for direct outbound connectivity to the internet. A NAT gateway is a fully managed, highly resilient NAT service that provides [scalable and on-demand SNAT](#).
 - Use a NAT gateway for:
 - Dynamic or large workloads sending traffic to the internet.
 - Static and predictable public IP addresses for outbound connectivity. NAT gateway can be associated with up to 16 public IP addresses or a /28 public IP prefix.
 - Mitigation of issues with SNAT port exhaustion commonly experienced with [Load balancer outbound rules](#), [Azure Firewall](#), or [Azure App Services](#).

- Security and privacy of resources within your network. Only outbound and return traffic can pass through NAT gateway.
- Use Azure Firewall to govern:
 - Azure outbound traffic to the internet.
 - Non-HTTP/S inbound connections.
 - East-west traffic filtering, if your organization requires it.
- Use [Azure Firewall Premium](#) for advanced firewall capabilities, such as:
 - Transport Layer Security (TLS) inspection.
 - A network intrusion detection and prevention system (IDPS).
 - URL filtering.
 - Web categories.
- [Azure Firewall Manager](#) supports both [Azure Virtual WAN](#) and regular virtual networks. Use Firewall Manager with Virtual WAN to deploy and manage Azure firewalls across Virtual WAN hubs or in hub virtual networks.
- If you use multiple IP addresses and ranges consistently in Azure Firewall rules, set up [IP Groups](#) in Azure Firewall. You can use the IP groups in Azure Firewall DNAT, network, and application rules for multiple firewalls across Azure regions and subscriptions.
- If you use a custom [user defined route](#) (UDR) to manage outbound connectivity to Azure platform as a service (PaaS) services, specify a [service tag](#) as the address prefix. Service tags update underlying IP addresses automatically to include changes, and reduce the overhead of managing Azure prefixes in a route table.
- Create a global Azure Firewall policy to govern security posture across the global network environment. Assign the policy to all Azure Firewall instances.
- Allow granular policies to meet specific region requirements by using Azure role-based access control to delegate incremental policies to local security teams.
- Use WAF within a landing-zone virtual network for protecting inbound HTTP/S traffic from the internet.
- Use Azure Front Door and WAF policies to provide global protection across Azure regions for inbound HTTP/S connections to a landing zone.
- To use Azure Front Door and Azure Application Gateway to help protect HTTP/S applications, use WAF policies in Azure Front Door. Lock down Azure Application Gateway to receive traffic only from Azure Front Door.

- If you need partner NVAs for inbound HTTP/S connections, deploy them within a landing-zone virtual network, together with the applications that they protect and expose to the internet.
- For outbound access, don't use Azure's default internet outbound access for any scenario. Issues experienced with default outbound access include:
 - Increased risk of SNAT port exhaustion.
 - Insecure by default.
 - Can't depend on default access IPs. They aren't owned by the customer and subject to change.
- Use a NAT gateway for online landing zones, or landing zones not connected to the hub virtual network. Compute resources that need outbound internet access and don't need the security of Azure Firewall standard or premium, or a third-party NVA, can use online landing zones.
- If your organization wants to use software-as-a-service (SaaS) security providers to help protect outbound connections, configure supported partners within Firewall Manager.
- If you use partner NVAs for east-west or north-south traffic protection and filtering:
 - For Virtual WAN network topologies, deploy the NVAs to a separate NVA virtual network. Connect the virtual network to the regional Virtual WAN hub and to the landing zones that need access to the NVAs. For more information, see [Scenario: Route traffic through an NVA](#).
 - For non-Virtual WAN network topologies, deploy the partner NVAs in the central hub virtual network.
- Don't expose VM management ports to the internet. For management tasks:
 - Use [Azure Policy](#) to prevent VM creation with public IPs.
 - Use [Azure Bastion](#) to access jumpbox VMs.
- Use [Azure DDoS Protection Standard](#) protection plans to help protect the public endpoints you host within your virtual networks.
- Don't try to replicate on-premises perimeter network concepts and architectures into Azure. Although Azure has similar security capabilities, the implementation and architecture are adapted to the cloud.

Plan for landing zone network segmentation

Article • 08/01/2024

This section explores key recommendations to deliver highly secure internal network segmentation within a landing zone to drive a network Zero Trust implementation.

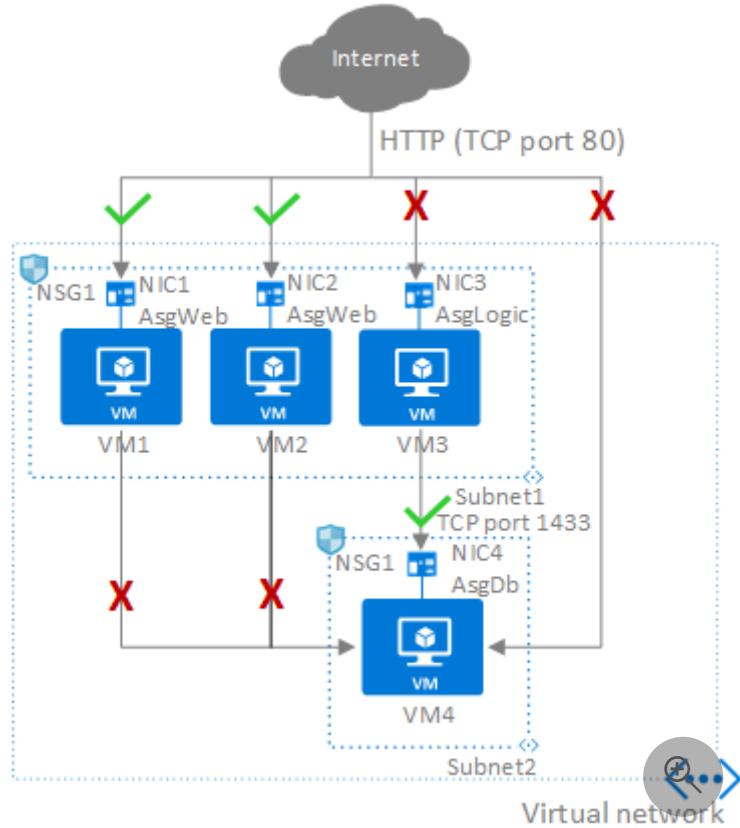
Design considerations

- The [Zero Trust model](#) assumes a breached state and verifies each request as though it originates from an uncontrolled network.
- An advanced Zero Trust network implementation employs fully distributed ingress and egress cloud micro-perimeters and deeper micro-segmentation.
- [Network security groups \(NSGs\)](#) can use Azure [service tags](#) to facilitate connectivity to Azure platform as a service (PaaS) solutions.
- [Application security groups \(ASGs\)](#) don't span or provide protection across virtual networks.
- Use [NSG flow logs](#) to inspect traffic that flows through a network point with an NSG attached.
- [Virtual network flow logs](#) provide capabilities that are similar to NSG flow logs but cover a wider range of use cases. They also simplify the scope of traffic monitoring because you can enable logging at the virtual network level.

Design recommendations

- Delegate subnet creation to the landing zone owner. This will enable them to define how to segment workloads across subnets (for example, a single large subnet, multitier application, or network-injected application). The platform team can use Azure Policy to ensure that an NSG with specific rules (such as deny inbound SSH or RDP from internet, or allow/block traffic across landing zones) is always associated with subnets that have deny-only policies.
- Use NSGs to help protect traffic across subnets, as well as east/west traffic across the platform (traffic between landing zones).

- The application team should use application security groups at the subnet-level NSGs to help protect multitier VMs within the landing zone.



- Use NSGs and application security groups to micro-segment traffic within the landing zone and avoid using a central NVA to filter traffic flows.
- Enable [virtual network flow logs](#) and use [traffic analytics](#) to gain insights into ingress and egress traffic flows. Enable flow logs on all critical virtual networks and subnets in your subscriptions, for example virtual networks and subnets that contain Windows Server Active Directory domain controllers or critical data stores. Additionally, you can use flow logs to detect and investigate potential security incidents, compliance and monitoring, and to optimize usage.
- Use NSGs to selectively allow connectivity between landing zones.
- For Virtual WAN topologies, route traffic across landing zones via Azure Firewall if your organization requires filtering and logging capabilities for traffic flowing across landing zones.
- If your organization decides to implement forced tunneling (advertise default route) to on-premises, we recommend incorporating the following **outbound** NSG rules to deny egress traffic from VNets directly to the internet should the BGP session drop.

Note

Rule priorities will need to be adjusted based on your existing NSG rule set.

[+] Expand table

Priority	Name	Source	Destination	Service	Action	Remark
100	AllowLocal	Any	VirtualNetwork	Any	Allow	Allow traffic during normal operations. With forced tunneling enabled, <code>0.0.0.0/0</code> is considered part of the <code>VirtualNetwork</code> tag as long as BGP is advertising it to the ExpressRoute or VPN Gateway.
110	DenyInternet	Any	Internet	Any	Deny	Deny traffic directly to the internet if the <code>0.0.0.0/0</code> route is withdrawn from the routes advertised (for example, due to an outage or misconfiguration).

✖ Caution

Azure PaaS services that can be injected into a virtual network maybe not compatible with forced tunneling. Control plane operations may still require direct connectivity to specific public IP addresses for the service to operate correctly. It's recommended to check the specific service documentation for networking requirements and eventually exempt the service subnet from the default route propagation. [Service Tags in UDR](#) can be used to bypass default route and redirect control plane traffic only, if the specific service tag is available.

Feedback

Was this page helpful?

👍 Yes

👎 No

Define network encryption requirements

Article • 08/01/2024

This section explores key recommendations to achieve network encryption between on-premises and Azure as well as across Azure regions.

Design considerations

- Cost and available bandwidth are inversely proportional to the length of the encryption tunnel between endpoints.
- [Azure Virtual Network encryption](#) enhances existing encryption-in-transit capabilities in Azure and allows seamless traffic encryption and decryption between virtual machines (VMs) and virtual machines scale sets.
- When you're using a VPN to connect to Azure, traffic is encrypted over the internet via IPsec tunnels.
- When you're using Azure ExpressRoute with private peering, traffic isn't currently encrypted.
- It's possible to configure a site-to-site [VPN connection over ExpressRoute private peering](#).
- You can apply [media access control security \(MACsec\)](#) encryption to ExpressRoute Direct to achieve network encryption.
- When Azure traffic moves between datacenters (outside physical boundaries not controlled by Microsoft or on behalf of Microsoft), [MACsec data-link layer encryption](#) is used on the underlying network hardware. This is applicable to virtual network peering traffic.

Design recommendations

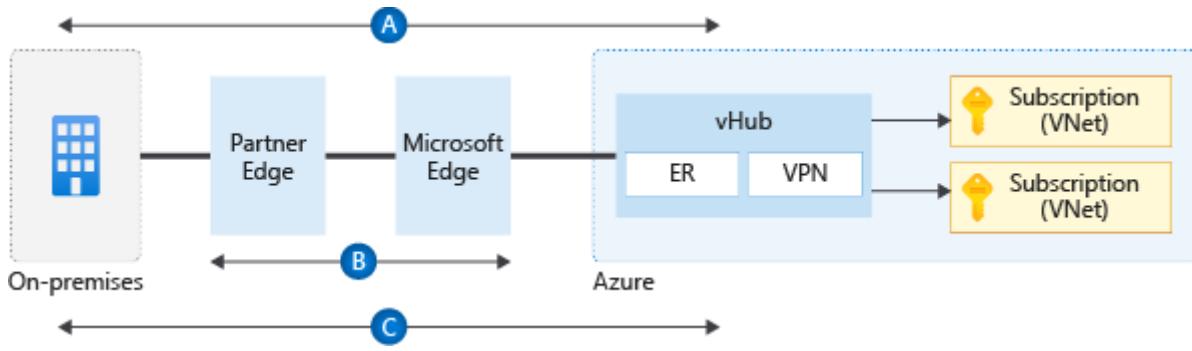


Figure 1: Encryption flows.

- When you're establishing VPN connections from on-premises to Azure by using VPN gateways, traffic is encrypted at a protocol level through IPsec tunnels. The preceding diagram shows this encryption in flow **A**.
- If you need to encrypt VM-to-VM traffic in the same virtual network or across regional or global peered virtual networks, use [Virtual Network encryption](#).
- When you're using ExpressRoute Direct, configure [MACsec](#) in order to encrypt traffic at Layer 2 between your organization's routers and MSEE. The diagram shows this encryption in flow **B**.
- For Virtual WAN scenarios where MACsec isn't an option (for example, not using ExpressRoute Direct), use a Virtual WAN VPN Gateway to establish [IPsec tunnels over ExpressRoute private peering](#). The diagram shows this encryption in flow **C**.
- For non-Virtual WAN scenarios, and where MACsec isn't an option (for example, not using ExpressRoute Direct), the only options are:
 - Use partner NVAs to establish IPsec tunnels over ExpressRoute private peering.
 - Establish a VPN tunnel over ExpressRoute with Microsoft peering.
 - Evaluate the capability to configure a Site-to-Site [VPN connection over ExpressRoute private peering](#).
- If native Azure solutions (as shown in flows **B** and **C** in the diagram) don't meet your requirements, use partner NVAs in Azure to encrypt traffic over ExpressRoute private peering.

Feedback

Was this page helpful?

Yes

No

Plan for traffic inspection

Article • 08/01/2024

Knowing what goes in and out of your network is essential to maintaining your security posture. You should capture all inbound and outbound traffic and perform near real-time analysis on that traffic to detect threats and mitigate network vulnerabilities.

This section explores key considerations and recommended approaches for capturing and analyzing traffic within an Azure virtual network.

Design considerations

Azure VPN Gateway: VPN Gateway lets you run a packet capture on a VPN gateway, a specific connection, multiple tunnels, one-way traffic, or bi-directional traffic. A maximum of five packet captures can run in parallel per gateway. They can be gateway-wide and per connection packet capture. For more information, see [VPN packet capture](#).

Azure ExpressRoute: You can use [Azure Traffic Collector](#) to gain visibility into traffic that traverses ExpressRoute circuits. To perform trending analysis, evaluate the amount of inbound and outbound traffic that goes through ExpressRoute. You can sample network flows that traverse the external interfaces of the Microsoft edge routers for ExpressRoute. A Log Analytics workspace receives the flow logs, and you can create your own log queries for further analysis. Traffic Collector supports both provider-managed circuits and ExpressRoute Direct circuits that have 1 Gbps or more bandwidth. Traffic Collector also supports private peering or Microsoft peering configurations.

Azure Network Watcher has multiple tools you should consider if you're using infrastructure as a service (IaaS) solutions:

- *Packet capture:* Network Watcher lets you create temporary capture packet sessions on traffic headed to and from a virtual machine. Each packet capture session has a time limit. When the session ends, packet capture creates a `pcap` file that you can download and analyze. Network Watcher packet capture can't give you continuous port mirroring with these time constraints. For more information, see [Packet capture overview](#).
- *Network security group (NSG) flow logs:* NSG flow logs capture information about IP traffic flowing through your NSGs. Network Watcher stores NSG flow logs as JSON files in Azure Storage account. You can export the NSG flow logs to an external tool for analysis. For more information, see NSG flow logs [overview](#) and [data analysis options](#).

- *Virtual network flow logs:* [Virtual network flow logs](#) provide similar capabilities compared to NSG flow logs. You can use virtual network flow logs to log information about Layer 3 traffic that flows through a virtual network. Azure Storage receives flow data from virtual network flow logs. You can access the data and export it to any visualization tool, security information and event management solution, or intrusion detection system.

Design recommendations

- Prefer [virtual network flow logs](#) over [NSG flow logs](#). Virtual network flow logs:
 - Simplify the scope of traffic monitoring. You can enable logging at the virtual network level so that you don't need to enable multiple-level flow logging to cover both subnet and NIC levels.
 - Add visibility for scenarios where you can't use NSG flow logs because of platform restrictions on NSG deployments.
 - Provide extra details about the [Virtual Network encryption status](#) and the presence of [Azure Virtual Network Manager security admin rules](#).

For a comparison, see [Virtual network flow logs compared to network security group flow logs](#).

- Don't enable virtual network flow logs and NSG flow logs simultaneously on the same target scope. If you enable NSG flow logs on the NSG of a subnet, and then you enable virtual network flow logs on the same subnet or parent virtual network, you duplicate logging and add extra costs.
- Enable traffic analytics. The tool lets you easily capture and analyze network traffic with out-of-the-box dashboard visualization and security analysis.
- If you need more capabilities than traffic analytics offers, you can supplement traffic analytics with one of our partner solutions. You can find available partner solutions in [Azure Marketplace](#) ↗.
- Use Network Watcher packet capture regularly to get a more detailed understanding of your network traffic. Run packet capture sessions at various times throughout the week to get a good understanding of the types of traffic traversing your network.
- Don't develop a custom solution to mirror traffic for large deployments. The complexity and supportability issues tend to make custom solutions inefficient.

Other platforms

- Manufacturing plants often have operational technology (OT) requirements that include traffic mirroring. Microsoft Defender for IoT can connect to a mirror on a switch or a terminal access point (TAP) for industrial control systems (ICS) or supervisory control and data acquisition (SCADA) data. For more information, see [traffic mirroring methods for OT monitoring](#).
- Traffic mirroring supports advanced workload deployment strategies in application development. With traffic mirroring, you can perform pre-production regression testing on live workload traffic or assess quality assurance and security assurance processes offline.
- When using Azure Kubernetes Service (AKS), ensure your ingress controller supports traffic mirroring if it's a part of your workload. Common ingress controllers that support traffic mirroring are [Istio](#), [NGINX](#), [Traefik](#).

Feedback

Was this page helpful?

 Yes

 No

Private Link and DNS integration at scale

Article • 10/25/2023

This article describes how to integrate Azure Private Link for PaaS services with Azure Private DNS zones in hub and spoke network architectures.

Introduction

Many customers build their network infrastructure in Azure using the hub and spoke network architecture, where:

- Networking shared services (such as network virtual appliances, ExpressRoute/VPN gateways, or DNS servers) deploy in the **hub** virtual network (VNet).
- **Spoke** VNets consume the shared services by way of VNet peering.

In hub and spoke network architectures, application owners are typically provided with an Azure subscription, which includes a VNet (*a spoke*) connected to the *hub* VNet. In this architecture, they can deploy their virtual machines and have private connectivity to other VNets or to on-premises networks by way of ExpressRoute or VPN.

A central network virtual appliance (NVA), such as Azure Firewall, provides Internet-outbound connectivity.

Many application teams build their solutions using a combination of Azure IaaS and PaaS resources. Some Azure PaaS services (such as SQL Managed Instance) can be deployed in customer VNets. As a result, traffic stays private within the Azure network and is fully routable from on-premises.

But some Azure PaaS services (such as Azure Storage or Azure Cosmos DB) can't be deployed in a customer's VNets and are accessible over their public endpoint. In some cases, this configuration causes a contention with a customer's security policies. Corporate traffic might not allow the deployment or accessing of corporate resources (such as a SQL database) over public endpoints.

Azure Private Link supports access to a [list of Azure services](#) over private endpoints, but it requires that you register those private endpoint records in a corresponding [private DNS zone](#).

This article describes how application teams can deploy Azure PaaS services in their subscriptions that are only accessible over private endpoints.

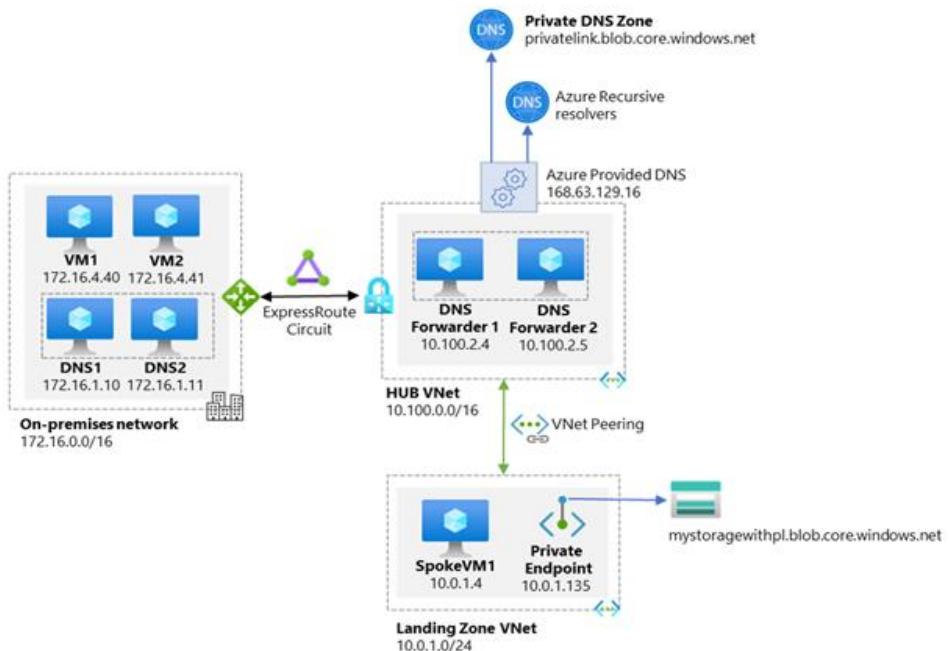
This article also describes how application teams can ensure that services automatically integrate with private DNS zones. They do the automation through Azure Private DNS, which removes the need to manually create or delete records in DNS.

Private Link and DNS integration in hub and spoke network architectures

Private DNS zones are typically hosted centrally in the same Azure subscription where the hub VNet deploys. This central hosting practice is driven by [cross-premises DNS name resolution](#) and other needs for central DNS resolution such as Active Directory. In most cases, only networking and identity administrators have permissions to manage DNS records in the zones.

Application teams have permissions to create Azure resource in their own subscription. They don't have any permissions in the central networking connectivity subscription, which includes managing DNS records in the private DNS zones. This access limitation means they don't have the ability to [create the DNS records required](#) when deploying Azure PaaS services with private endpoints.

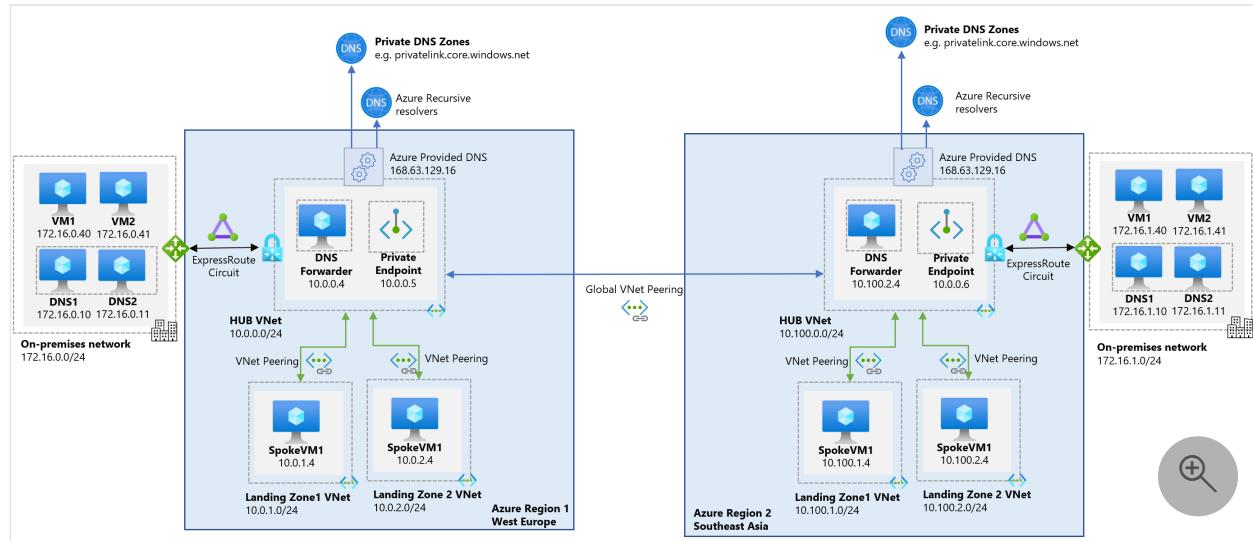
The following diagram shows a typical high-level architecture for enterprise environments with central DNS resolution and where name resolution for Private Link resources is done via Azure Private DNS:



From the previous diagram, it's important to highlight that:

- On-premises DNS servers have conditional forwarders configured for each private endpoint public DNS zone, pointing to the DNS servers `10.100.2.4` and `10.100.2.5` hosted in the hub VNet.
- The DNS servers `10.100.2.4` and `10.100.2.5` hosted in the hub VNet use the Azure-provided DNS resolver (`168.63.129.16`) as a forwarder.
- The hub VNet must be linked to the Private DNS zone names for Azure services (such as `privatelink.blob.core.windows.net`, as shown in the diagram).
- All Azure VNets use the DNS servers hosted in the hub VNet (`10.100.2.4` and `10.100.2.5`) as the primary and secondary DNS servers.
- If the DNS servers `10.100.2.4` and `10.100.2.5` aren't authoritative for customer's corporate domains (for example, Active Directory domain names), they should have conditional forwarders for the customer's corporate domains, pointing to the on-premises DNS Servers (`172.16.1.10` and `172.16.1.11`) or DNS servers deployed in Azure that are authoritative for such zones.

While the previous diagram depicts a single hub and spoke architecture, customers might need to extend their Azure footprint across multiple regions to address resiliency, proximity or data residency requirements, several scenarios have emerged where the same Private-Link-enabled PaaS instance must be accessed through multiple Private Endpoints (PE's).



The following diagram shows a typical high-level architecture for enterprise environments with central DNS resolution deployed in the hub (one per region) where name resolution for Private Link resources is done via Azure Private DNS.

It is recommended to deploy multiple regional private endpoints associated to the PaaS instance, one in each region where clients exist, enable per-region Private Link and Private DNS Zones. When working with PaaS services with built-in DR capabilities (geo-

redundant storage accounts, SQL DB failover groups, etc.), multiple region Private Endpoints are mandatory.

This scenario requires manual maintenance/updates of the Private Link DNS record set in every region as there is currently no automated lifecycle management for these.

For other use cases, a single global Private Endpoint can be deployed, making accessible to all clients by adding routing from the relevant regions to the single Private Endpoint in a single region.

To enable resolution, and therefore connectivity, from on premise networks to the `privatelink` private DNS zone and private endpoints, the appropriate DNS configuration (conditional forwarders etc.) need to be provisioned in the DNS infrastructure.

There are two conditions that must be true for application teams to create any required Azure PaaS resources in their subscription:

- Central networking and/or central platform team must ensure that application teams can only deploy and access Azure PaaS services by way of private endpoints.
- Central networking and/or central platform teams must ensure that when they create private endpoints, they set up how to handle the corresponding records. Set up the corresponding records such that they're automatically created in the centralized private DNS zone that matches the service being created.
- DNS records must follow the lifecycle of the private endpoint, in that, it's automatically removed when the private endpoint is deleted.

ⓘ Note

if FQDNs in network rules based on DNS resolution is needed to be used in Azure Firewall and Firewall policy (This capability allows you to filter outbound traffic with any TCP/UDP protocol -including NTP, SSH, RDP, and more-). You must enable Azure Firewall DNS Proxy to use FQDNs in your network rules, then those spoke VNets are forced to change their DNS setting from custom DNS server to Azure Firewall DNS Proxy. Changing the DNS settings of a spoke VNet requires reboot of all VMs inside that VNet.

The following sections describe how application teams enable these conditions by using [Azure Policy](#). The example uses Azure Storage as the Azure service that application teams need to deploy. But the same principle applies to most [Azure services that support Private Link](#).

Configuration required by the platform team

The platform team configuration requirements include creating the private DNS zones, setting up policy definitions, deploying policies, and setting up the policy assignments.

Create private DNS zones

Create private DNS zones in the central connectivity subscription for the supported Private Link services. For more information, see [Azure Private Endpoint DNS configuration](#).

In this case, **Storage account with blob** is the example. It translates to creating a `privatelink.blob.core.windows.net` private DNS zone in the connectivity subscription.

Name ↑↓	Type ↑↓	Location ↑↓
privatelink.blob.core.windows.net	Private DNS zone	Global

Policy definitions

In addition to the private DNS zones, you also need to [create a set of custom Azure Policy definitions](#). These definitions enforce the use of private endpoints and automate creating the DNS record in the DNS zone that you create:

1. Deny public endpoint for PaaS services policy.

This policy prevents users from creating Azure PaaS services with public endpoints and gives them an error message if they don't select the private endpoint when creating the resource.

Create storage account ...

Basics Networking Data protection Advanced Tags Review + create

Network connectivity

You can connect to your storage account either publicly, via public IP addresses or service endpoints, or privately, using a private endpoint.

Connectivity method *

Public endpoint (all networks)
 Public endpoint (selected networks)
 Private endpoint

i All networks will be able to access this storage account.
[Learn more about connectivity methods](#)

Create storage account

 Validation failed. Click here to view details. →

Basics Networking Data protection Advanced Tags **Review + create**

Summary Raw Error

ERROR DETAILS



Resource 'mystorageaccountwithpl' was disallowed by policy. (Code: RequestDisallowedByPolicy)

Initiative: Deny-Public-Endpoints-for-PaaS-Services
Policy: Deny-PublicEndpoint-Storage

The exact policy rule might differ between PaaS services. For Azure Storage accounts, look at the **networkAcls.defaultAction** property that defines whether requests from public networks are allowed or not. In this case, set a condition to deny creating the **Microsoft.Storage/storageAccounts** resource type if the property **networkAcls.defaultAction** isn't `Deny`. The following policy definition shows the behavior:

JSON

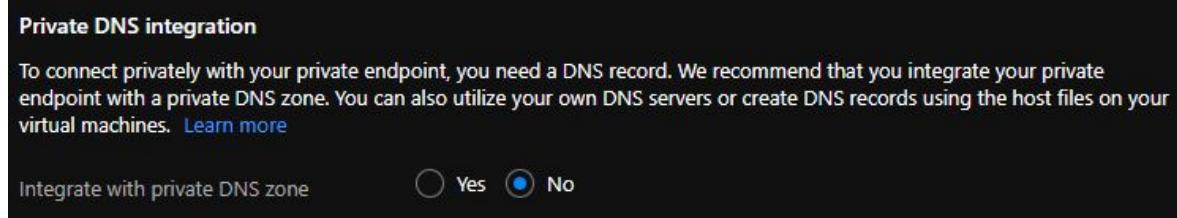
```
{  
  "mode": "All",  
  "policyRule": {  
    "if": {  
      "allOf": [  
        {  
          "field": "type",  
          "equals": "Microsoft.Storage/storageAccounts"  
        },  
        {  
          "field":  
            "Microsoft.Storage/storageAccounts/networkAcls.defaultAction",  
          "notEquals": "Deny"  
        }  
      ]  
    },  
    "then": {  
      "effect": "Deny"  
    }  
  }  
}
```

```
}
```

- Deny the ability to create a private DNS zone with the `privatelink` prefix policy.

Use a centralized DNS architecture with a conditional forwarder and private DNS zones hosted in the subscriptions managed by the platform team. It's necessary to prevent the application teams owners from creating their own Private Link private DNS zones and linking services into their subscriptions.

Ensure that when your application team creates a private endpoint, the option to `Integrate with private DNS zone` is set to `No` in the Azure portal.



If you select `Yes`, Azure Policy prevents you from creating the private endpoint. In the policy definition, it denies the ability to create the `Microsoft.Network/privateDnsZones` resource type if the zone has the `privatelink` prefix. The following policy definition shows the `privatelink` prefix:

JSON

```
{
  "description": "This policy restricts creation of private DNS zones with the `privatelink` prefix",
  "displayName": "Deny-PrivateDNSZone-PrivateLink",
  "mode": "All",
  "parameters": null,
  "policyRule": {
    "if": {
      "allOf": [
        {
          "field": "type",
          "equals": "Microsoft.Network/privateDnsZones"
        },
        {
          "field": "name",
          "contains": "privatelink."
        }
      ]
    },
    "then": {
      "effect": "Deny"
    }
  }
}
```

```
}
```

3. `DeployIfNotExists` policy to automatically create the required DNS record in the central private DNS zone.

The following policy examples show two approaches for identifying which `privateDNSZoneGroup` is created on a Private Endpoint.

The [first policy](#) relies on the `groupId` while the [second policy](#) uses both `privateLinkServiceId` and `groupID`. Use the [second policy](#) when `groupId` will clash (collide) with another resource.

For example, the `groupId` SQL is used for both Cosmos DB and Synapse Analytics. If both resource types deploy and the [first policy](#) has been assigned to create the `privateDNSZoneGroup` on the Private Endpoint entry, it's created and mapped to the incorrect Private DNS Zone, of either Cosmos DB or Synapse Analytics. It then might toggle between each of the zones due to the clashing `groupId` that the first policy looks for in its policy rule.

For a list of Private-link resources `groupId`, see the subresources column in [What is a private endpoint?](#).

Tip

Azure Policy built-in definitions are constantly being added, deleted, and updated. It's highly recommended to use built-in policies versus managing your own policies (where available). Use the [AzPolicyAdvertiser](#) to find existing built-in policies that have the following name of 'xxx ... to use private DNS zones'. In addition, Azure Landing Zones (ALZ) has a policy initiative, [Configure Azure PaaS services to use private DNS zones](#), that contains built-in policies as well and periodically updated. If a built-in policy isn't available for your situation, consider creating an issue on the `azure-policy` feedback site [Azure Governance · Community](#) following the [New built-in Policy Proposals process on the Azure Policy GitHub repo.](#)

First `DeployIfNotExists` Policy - Matching on `groupId` only

This policy triggers if you create a private endpoint resource with a service-specific `groupId`. The `groupId` is the ID of the group obtained from the remote resource (service) that this private endpoint should connect to. It then triggers a deployment of a

`privateDNSZoneGroup` within the private endpoint, which associates the private endpoint with the private DNS zone. In the example, the `groupId` for Azure Storage blobs is `blob`. For more information on the `groupId` for other Azure services, see [Azure Private Endpoint DNS configuration](#), under the **Subresource** column. When the policy finds the `groupId` in the private endpoint, it deploys a `privateDNSZoneGroup` within the private endpoint, and links it to the private DNS zone resource ID that's specified as the parameter. In the example, the private DNS zone resource ID is:

```
/subscriptions/<subscription-id>/resourceGroups/<resourceGroupName>/providers/Microsoft.Network/privateDnsZones/privatelink.blob.core.windows.net
```

The following code sample shows the policy definition:

JSON

```
{
  "mode": "Indexed",
  "policyRule": {
    "if": {
      "allOf": [
        {
          "field": "type",
          "equals": "Microsoft.Network/privateEndpoints"
        },
        {
          "count": {
            "field":
              "Microsoft.Network/privateEndpoints/privateLinkServiceConnections[*].groupId
              s[*]",
            "where": {
              "field":
                "Microsoft.Network/privateEndpoints/privateLinkServiceConnections[*].groupId
                s[*]",
              "equals": "blob"
            }
          }
        },
        {
          "greaterOrEquals": 1
        }
      ]
    },
    "then": {
      "effect": "deployIfNotExists",
      "details": {
        "type": "Microsoft.Network/privateEndpoints/privateDnsZoneGroups",
        "roleDefinitionIds": [
          "/providers/Microsoft.Authorization/roleDefinitions/4d97b98b-1d4f-
          4787-a291-c67834d212e7"
        ],
        "deployment": {
          "name": "DeployPrivateDnsZoneGroup"
        }
      }
    }
  }
}
```

```
"properties": {
    "mode": "incremental",
    "template": {
        "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
        "contentVersion": "1.0.0.0",
        "parameters": {
            "privateDnsZoneId": {
                "type": "string"
            },
            "privateEndpointName": {
                "type": "string"
            },
            "location": {
                "type": "string"
            }
        },
        "resources": [
            {
                "name": "[concat(parameters('privateEndpointName'), '/deployedByPolicy')]",
                "type":
"Microsoft.Network/privateEndpoints/privateDnsZoneGroups",
                "apiVersion": "2020-03-01",
                "location": "[parameters('location')]",
                "properties": {
                    "privateDnsZoneConfigs": [
                        {
                            "name": "storageBlob-privateDnsZone",
                            "properties": {
                                "privateDnsZoneId": "
[parameters('privateDnsZoneId')]"
                            }
                        }
                    ]
                }
            },
            "parameters": {
                "privateDnsZoneId": {
                    "value": "[parameters('privateDnsZoneId')]"
                },
                "privateEndpointName": {
                    "value": "[field('name')]"
                },
                "location": {
                    "value": "[field('location')]"
                }
            }
        ]
    }
},
```

```

"parameters": {
    "privateDnsZoneId": {
        "type": "String",
        "metadata": {
            "displayName": "privateDnsZoneId",
            "strongType": "Microsoft.Network/privateDnsZones"
        }
    }
}

```

Second `DeployIfNotExists` Policy - Matching on `groupId` & `privateLinkServiceId`

This policy triggers if you create a private endpoint resource with a service-specific `groupId` and `privateLinkServiceId`. The `groupId` is the ID of the group obtained from the remote resource (service) that this private endpoint should connect to. The `privateLinkServiceId` is the resource ID of the remote resource (service) this private endpoint should connect to. Then, trigger a deployment of a `privateDNSZoneGroup` within the private endpoint, which associates the private endpoint with the private DNS zone.

In the example, the `groupId` for Azure Cosmos DB (SQL) is `SQL` and the `privateLinkServiceId` must contain `Microsoft.DocumentDb/databaseAccounts`. For more information on the `groupId` and `privateLinkServiceId` for other Azure services, see [Azure Private Endpoint DNS configuration](#), under the **Subresource** column. When the policy finds `groupId` and `privateLinkServiceId` in the private endpoint, it deploys a `privateDNSZoneGroup` within the private endpoint. And it's linked to the private DNS zone resource ID that's specified as the parameter. The following policy definition shows the private DNS zone resource ID:

```

/subscriptions/<subscription-
id>/resourceGroups/<resourceGroupName>/providers/Microsoft.Network/privateDnsZones/
privatelink.documents.azure.com

```

The following code sample shows the policy definition:

JSON

```
{
    "mode": "Indexed",
    "policyRule": {
        "if": {
            "allOf": [
                {

```

```
        "field": "type",
        "equals": "Microsoft.Network/privateEndpoints"
    },
{
    "count": {
        "field":
"Microsoft.Network/privateEndpoints/privateLinkServiceConnections[*]",
        "where": {
            "allOf": [
                {
                    "field":
"Microsoft.Network/privateEndpoints/privateLinkServiceConnections[*].private
LinkServiceId",
                    "contains": "Microsoft.DocumentDb/databaseAccounts"
                },
                {
                    "field":
"Microsoft.Network/privateEndpoints/privateLinkServiceConnections[*].groupId
s[*]",
                    "equals": "[parameters('privateEndpointGroupId')]"
                }
            ]
        }
    },
    "greaterOrEquals": 1
}
],
{
    "then": {
        "effect": "[parameters('effect')]",
        "details": {
            "type": "Microsoft.Network/privateEndpoints/privateDnsZoneGroups",
            "roleDefinitionIds": [
                "/providers/Microsoft.Authorization/roleDefinitions/4d97b98b-1d4f-
4787-a291-c67834d212e7"
            ],
            "deployment": {
                "properties": {
                    "mode": "incremental",
                    "template": {
                        "$schema": "https://schema.management.azure.com/schemas/2019-
04-01/deploymentTemplate.json#",
                        "contentVersion": "1.0.0.0",
                        "parameters": {
                            "privateDnsZoneId": {
                                "type": "string"
                            },
                            "privateEndpointName": {
                                "type": "string"
                            },
                            "location": {
                                "type": "string"
                            }
                        },
                        "resources": [

```

```

        {
            "name": "[concat(parameters('privateEndpointName'),
'/_deployedByPolicy')]",
            "type": "Microsoft.Network/privateEndpoints/privateDnsZoneGroups",
            "apiVersion": "2020-03-01",
            "location": "[parameters('location')]",
            "properties": {
                "privateDnsZoneConfigs": [
                    {
                        "name": "cosmosDB-privateDnsZone",
                        "properties": {
                            "privateDnsZoneId": "[parameters('privateDnsZoneId')]"
                        }
                    }
                ]
            }
        },
        "parameters": {
            "privateDnsZoneId": {
                "value": "[parameters('privateDnsZoneId')]"
            },
            "privateEndpointName": {
                "value": "[field('name')]"
            },
            "location": {
                "value": "[field('location')]"
            }
        }
    },
    "parameters": {
        "privateDnsZoneId": {
            "type": "String",
            "metadata": {
                "displayName": "Private Dns Zone Id",
                "description": "The private DNS zone to deploy in a new private DNS zone group and link to the private endpoint",
                "strongType": "Microsoft.Network/privateDnsZones"
            }
        },
        "privateEndpointGroupId": {
            "type": "String",
            "metadata": {
                "displayName": "Private Endpoint Group Id",
                "description": "A group Id for the private endpoint"
            }
        },
        "effect": {

```

```
        "type": "String",
        "metadata": {
            "displayName": "Effect",
            "description": "Enable or disable the execution of the policy"
        },
        "allowedValues": [
            "DeployIfNotExists",
            "Disabled"
        ],
        "defaultValue": "DeployIfNotExists"
    }
}
}
```

Policy assignments

After policy definitions are deployed, [assign the policies](#) at the desired scope in your management group hierarchy. Ensure that the policy assignments target the Azure subscriptions the application teams use to deploy PaaS services with private endpoint access exclusively.

ⓘ Important

In addition to [assigning the roleDefinition](#) defined in the policy, remember to assign the [Private DNS Zone Contributor role](#) role in the subscription and resource group where the private DNS zones are hosted to the [managed identity created by the DeployIfNotExists policy assignment](#) that will be responsible to create and manage the private endpoint DNS record in the private DNS zone. This is because the private endpoint is located in the application owner Azure subscription, while the private DNS zone is located in a different subscription (such as central connectivity subscription).

After the platform team finishes the configuration:

- The applications teams' Azure subscriptions are ready for the team to then create Azure PaaS services with private endpoint access exclusively.
- The team must ensure the DNS records for private endpoints are automatically registered (and removed once a private endpoint is deleted) from the corresponding private DNS zones.

Application owner experience

After the platform team deploys the platform infrastructure components (private DNS zones and policies), the application owner has the following experience when they try to deploy an Azure PaaS service into the Azure subscription. This experience is the same whether they do their activities through the Azure portal or other clients, such as PowerShell or CLI, since Azure policies govern their subscriptions.

1. Create a storage account through the Azure portal. In the **Basics** tab, choose the settings you want, provide a name for your storage account, and select **Next**.

The screenshot shows the 'Create storage account' wizard in the Azure portal, specifically the 'Basics' tab. The page title is 'Create storage account' with a '...' button. Below the title is a navigation bar with tabs: Basics (which is selected and underlined), Networking, Data protection, Advanced, Tags, and Review + create. A descriptive text block states: 'Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below.' It also includes a link 'Learn more about Azure storage accounts'. The 'Project details' section asks to select a subscription and resource group. The 'Subscription' dropdown is set to 'Azure Internal Subscription'. The 'Resource group' dropdown has 'storage' selected and 'Create new' option available. The 'Instance details' section asks for the storage account name and location. The 'Storage account name' input field contains 'mystoragewithpl' with a green checkmark icon. The 'Location' dropdown is set to '(Europe) North Europe'. The entire form is contained within a dark-themed card.

2. In the networking tab, select **Private endpoint**. If you select an option other than **Private endpoint**, the Azure portal won't allow you to create the storage account in the **Review + create** section of the deployment wizard. The policy prevents you from creating this service if the public endpoint is enabled.

Create storage account

Basics Networking Data protection Advanced Tags Review + create

Network connectivity

You can connect to your storage account either publicly, via public IP addresses or service endpoints, or privately, using a private endpoint.

Connectivity method *

- Public endpoint (all networks)
- Public endpoint (selected networks)
- Private endpoint

Private endpoint

Create a private endpoint to allow a private connection to this resource. Additional private endpoint connections can be created. [Learn more about private endpoints](#)

Name	Subscription	Resource group	Region	Target sub-resource
Click on add to create a private endpoint				

[Add](#)

Network routing

Determine how to route your traffic as it travels from the source to its Azure endpoint. Microsoft network routing is recommended.

Routing preference *

- Microsoft network routing (default)
- Internet routing

[Review + create](#)

[< Previous](#)

[Next : Data protection >](#)

3. It's possible to create the private endpoint now or after you create the storage account. This example shows creating the private endpoint after the storage account is created. Select **Review + create** to complete the step.

4. After you create the storage account, make a private endpoint through the Azure portal.

Create a private endpoint

1 Basics

2 Resource

3 Configuration

4 Tags

5 Review + create

Use private endpoints to privately connect to a service or resource. Your private endpoint must be in the same region as your virtual network, but can be in a different region from the private link resource that you are connecting to. [Learn more](#)

Project details

Subscription * ⓘ

Azure Internal Subscription

Resource group * ⓘ

storage

[Create new](#)

Instance details

Name *

mystoragewithpl-pe

Region *

(Europe) North Europe

5. In the **Resource** section, locate the storage account you created in the previous step. Under target subresource, select **Blob**, and then select **Next**.

Create a private endpoint

✓ Basics 2 Resource 3 Configuration 4 Tags 5 Review + create

Private Link offers options to create private endpoints for different Azure resources, like your private link service, a SQL server, or an Azure storage account. Select which resource you would like to connect to using this private endpoint. [Learn more](#)

Connection method ⓘ Connect to an Azure resource in my directory. Connect to an Azure resource by resource ID or alias.

Subscription * ⓘ

Resource type * ⓘ

Resource * ⓘ

Target sub-resource * ⓘ

6. In the **Configuration** section, after selecting your VNet and subnet, be sure that **Integrate with private DNS zone** is set to **No**. Otherwise, the Azure portal prevents you from creating the private endpoint. Azure Policy won't allow you to create a private DNS zone with the `privatelink` prefix.

✓ Basics ✓ Resource 3 Configuration 4 Tags 5 Review + create

Networking

To deploy the private endpoint, select a virtual network subnet. [Learn more](#)

Virtual network * ⓘ

Subnet * ⓘ

ⓘ If you have a network security group (NSG) enabled for the subnet above, it will be disabled for private endpoints on this subnet only. Other resources on the subnet will still have NSG enforcement.

Private DNS integration

To connect privately with your private endpoint, you need a DNS record. We recommend that you integrate your private endpoint with a private DNS zone. You can also utilize your own DNS servers or create DNS records using the host files on your virtual machines. [Learn more](#)

Integrate with private DNS zone Yes No

7. Select **Review + create**, and then select **Create** to deploy the private endpoint.
8. After a few minutes, the `DeployIfNotExists` policy triggers. The subsequent `dnsZoneGroup` deployment then adds the required DNS records for the private endpoint in the centrally managed DNS zone.
9. After you create the private endpoint, select it, and review its FQDN and private IP:

mystoragewithpl-pe | DNS configuration

Private endpoint | Directory: Microsoft

Search (Ctrl+ /) < Add configuration Refresh

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Settings

DNS configuration Properties Locks

Private DNS integration To connect privately with your private endpoint, you need a DNS record. We recommend that you integrate your private endpoint using a private DNS zone. You can also utilize your own DNS servers. [Learn more](#)

Custom DNS records To be configured correctly, the following FQDNs are required in your private DNS setup. [Learn more](#)

FQDN	IP addresses
mystoragewithpl.blob.core.windows.net	10.1.0.132

10. Check the activity log for the resource group where the private endpoint was created. Or you can check the activity log of the private endpoint itself. You'll notice that after a few minutes, a `DeployIfNotExist` policy action runs and that configures the DNS zone group on the private endpoint:

mystoragewithpl-pe | Activity log

Private endpoint | Directory: Microsoft

Search (Ctrl+ /) < Activity Edit columns Refresh Diagnostics settings Download as CSV Logs Pin current filters Reset filters

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Settings

DNS configuration Properties Locks

Monitoring

Alerts Metrics

Automation

Tasks (preview) Export template

Management Group : None Subscription : Azure Internal Subscription Event severity : All Timespan : Last 6 hours Resource

+ Add Filter

11 items.

Operation name	Status	Time	Time stamp	Subscription
DeployIfNotExists' Policy action.	Succeeded	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription
DeployIfNotExists	Accepted	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription
'DeployIfNotExists' Policy action.	Accepted	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription
Put Private DNS Zone Group	Started	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription
Put Private DNS Zone Group	Started	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription
Put Private DNS Zone Group	Accepted	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription
Put Private DNS Zone Group	Accepted	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription
Write PrivateDnsZoneGroups	Succeeded	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription
Put Private DNS Zone Group	Succeeded	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription
DeployIfNotExists	Succeeded	22 minutes ...	Tue Mar 09 ...	Azure Internal Subscription

11. If the central networking team goes to the `privatelink.blob.core.windows.net` private DNS zone, they'll confirm that the DNS record is there for the private endpoint you created, and both the name and IP address match the values within the private endpoint.

The screenshot shows the Azure portal interface for managing a private DNS zone. The left sidebar has sections for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings (Virtual network links, Properties, Locks, Monitoring, Alerts, Metrics), and a search bar for record sets. The main area displays the 'Essentials' section with fields for Resource group, Subscription, Subscription ID, and Tags. Below this is a table of record sets:

Name	Type	TTL	Value
@	SOA	3600	Email: azureprivatedns-host.microsoft.com Host: azureprivatedns.net Refresh: 3600 Retry: 300 Expire: 2419200 Minimum TTL: 10 Serial number: 1
mystoragewithpl	A	10	10.1.0.132

At this point, application teams can use the storage account through a private endpoint from any VNet in the hub and spoke network environment and from on-premises. The DNS record has been automatically recorded in the private DNS zone.

If an application owner deletes the private endpoint, the corresponding records in the private DNS zone are automatically removed.

Next steps

Review [DNS for on-premises and Azure resources](#). Review [Plan for virtual machine remote access](#).

ⓘ Important

This article outlines DNS and Private link integration at scale using DINE (DeployIfNotExists) policies assigned to the Management Group. Which means there is no need to handle the DNS integration in code when creating Private Endpoints with this approach, as it is handled by the policies. It is also unlikely that the application teams have RBAC access to the centralized Private DNS Zones also.

Below are helpful links to review when creating Private Endpoint with Bicep and HashiCorp Terraform.

For Private Endpoint creation with Infrastructure-as-Code:

- [Quickstart Create a private endpoint using Bicep](#).
- Create a private endpoint using HashiCorp Terraform [azurerm_private_endpoint](#) in Terrafrom Registry.

You can still create private endpoints in your Infrastructure-as-Code tooling however, if using the DINE policy approach as outlined in this article you should leave the DNS integration side out of your code and let the DINE policies that have the required RBAC to the Private DNS Zones handle this instead.

DNS for on-premises and Azure resources

Article • 06/08/2023

Domain Name System (DNS) is a critical design topic in the overall landing zone architecture. Some organizations might want to use their existing investments in DNS. Others might see cloud adoption as an opportunity to modernize their internal DNS infrastructure and use native Azure capabilities.

Design considerations:

- You can use Azure DNS Private Resolver service in conjunction with Azure Private DNS Zones for cross-premises name resolution.
- You might require the use of existing DNS solutions across on-premises and Azure.
- The maximum number of private DNS zones a virtual network can be linked to with auto-registration enabled is one.
- Familiarize yourself with [Azure Private DNS zone limits](#).

Design recommendations:

- For environments where name resolution in Azure is all that's required, use Azure Private DNS zones for resolution. Create a delegated zone for name resolution (such as `azure.contoso.com`). Enable auto-registration for Azure Private DNS zone to automatically manage the lifecycle of the DNS records for the virtual machines deployed within a virtual network.
- For environments where name resolution across Azure and on-premises is required, it is recommended to use DNS Private Resolver service along with Azure Private DNS Zones. It offers many benefits over virtual machines based DNS solution, including cost reduction, built-in high availability, scalability, and flexibility.

If you need to use existing DNS infrastructure (for example, Active Directory integrated DNS), ensure that the DNS server role is deployed onto at least two VMs and configure DNS settings in virtual networks to use those custom DNS servers.

- For environments with Azure Firewall, consider using it as [DNS proxy](#).

- You can link an Azure Private DNS zone to the virtual networks and use DNS Private Resolver service with DNS forwarding rule set also associated with the virtual networks:
 - For DNS queries generated in the Azure virtual network to resolve on-premises DNS names such as `corporate.contoso.com`, the DNS query is forwarded to the IP address of on-premises DNS servers specified in the rule set.
 - For DNS queries generated in the on-premises network to resolve DNS records in Azure Private DNS Zones, you can configure on-premises DNS servers with conditional forwarders pointing to DNS Private Resolver service's inbound endpoint IP address in Azure, to forward the request to the Azure Private DNS zone (for example, `azure.contoso.com`).
- Special workloads that require and deploy their own DNS (such as Red Hat OpenShift) should use their preferred DNS solution.
- Create the Azure Private DNS zones within a global connectivity subscription. The Azure Private DNS zones that should be created include the zones required for accessing Azure PaaS services via a **private endpoint** (for example, `privatelink.database.windows.net` or `privatelink.blob.core.windows.net`).

Plan for virtual machine remote access

Article • 07/05/2023

This article describes the recommended guidance for providing remote access to virtual machines (VMs) deployed within an Azure landing zones architecture.

Azure offers different technologies for providing remote access to VMs:

- [Azure Bastion](#), a platform as a service (PaaS) solution, for accessing VMs through a browser or currently in preview through the native SSH/RDP client on Windows workstations
- [Just in time \(JIT\)](#) access provided through Microsoft Defender for Cloud
- Hybrid connectivity options, such as Azure ExpressRoute and VPNs
- Public IP attached directly to the VM or through a NAT rule via an Azure public load balancer

The choice of which remote access solution is most appropriate depends on factors like scale, topology, and security requirements.

Design considerations

- When available, you can use existing hybrid connectivity to Azure virtual networks via ExpressRoute or S2S/P2S VPN connections to provide remote access from on-premises to Windows and Linux Azure VMs.
- NSGs can be used to secure SSH/RDP connections to Azure VMs.
- JIT allows remote SSH/RDP access over the internet without having to deploy any other infrastructure.
- There are some [availability limitations](#) with JIT access.
 - JIT access can't be used for VMs protected by Azure firewalls controlled by Azure Firewall Manager.
- [Azure Bastion](#) provides an extra layer of control. It enables secure and seamless RDP/SSH connectivity to your VMs directly from the Azure portal or [native client](#) in preview over a secure TLS channel. Azure Bastion also negates the need for hybrid connectivity.
- Consider the appropriate Azure Bastion SKU to use based on your requirements as described in [About Azure Bastion configuration settings](#).
- Review the [Azure Bastion FAQ](#) for answers to common questions you might have about the service.
- Azure Bastion can be used in [Azure Virtual WAN topology](#) however there are some limitations:

- Azure Bastion cannot be deployed inside of a Virtual WAN virtual hub.
- Azure Bastion must use the **Standard** SKU and also the **IP based connection** feature must be enabled on the Azure Bastion resource, see the [Azure Bastion IP based connection documentation](#)
- Azure Bastion can be deployed in any spoke virtual network connected in a Virtual WAN, for accessing Virtual Machines, in its own or, other virtual networks that are connected to the same Virtual WAN, via its associated hubs, through Virtual WAN virtual network connections; providing **routing** is configured correctly.

Tip

Azure Bastion IP based connection also allows for connectivity to on-premises based machines, providing there is hybrid connectivity established between the Azure Bastion resource and the machine you are wanting to connect to. See, [Connect to a VM via specified private IP address through the portal](#)

Design recommendations

- Use existing ExpressRoute or VPN connectivity to provide remote access to Azure VMs that are accessible from on-premises via ExpressRoute or VPN connections.
- In a Virtual WAN-based network topology where remote access to Virtual Machines over the internet is required:
 - Azure Bastion can be deployed in each spoke virtual network of the respective VMs.
 - Or you may choose to deploy a centralized Azure Bastion instance in a single spoke in your Virtual WAN topology, as shown in Figure 1. This configuration reduces the number of Azure Bastion instances to manage in your environment. This scenario requires users who sign in to Windows and Linux VMs via Azure Bastion to have a [reader role on the Azure Bastion resource and the chosen spoke virtual network](#). Some implementations might have security or compliance considerations that restrict or prevent this.
- In hub-and-spoke network topology, where remote access to Azure Virtual Machines over the internet is required:
 - A single Azure Bastion host can be deployed in the hub virtual network, which can provide connectivity to Azure VMs on spoke virtual networks via virtual network peering. This configuration reduces the number of Azure Bastion instances to manage in your environment. This scenario requires users who sign in to Windows and Linux VMs via Azure Bastion to have a [reader role on the](#)

Azure Bastion resource and the hub virtual network. Some implementations might have security or compliance considerations. See Figure 2.

- Your environment might not permit granting users the reader role-based access control (RBAC) role on the Azure Bastion resource and the hub virtual network. Use Azure Bastion Basic or Standard to provide connectivity to VMs within a spoke virtual network. Deploy a dedicated Azure Bastion instance into each spoke virtual network that requires remote access. See Figure 3.
- Configure NSG rules to protect Azure Bastion and the VMs to which it provides connectivity. Follow the guidance in [Working with VMs and NSGs in Azure Bastion](#).
- Configure Azure Bastion diagnostic logs to be sent to the central Log Analytics workspace. Follow the guidance in [Enable and work with Azure Bastion resource logs](#).
- Ensure the [required RBAC role assignments](#) are made for the users or groups that connect to the VMs via Azure Bastion are in place.
- If you connect to Linux VMs via SSH, use the feature of [connecting by using a private key stored in Azure Key Vault](#).
- Deploy Azure Bastion and ExpressRoute or VPN access to address specific needs like emergency break-glass access.
- Remote access to Windows and Linux VMs via public IPs directly attached to the VMs is discouraged. Remote access should never be deployed without strict NSG rules and firewalling.

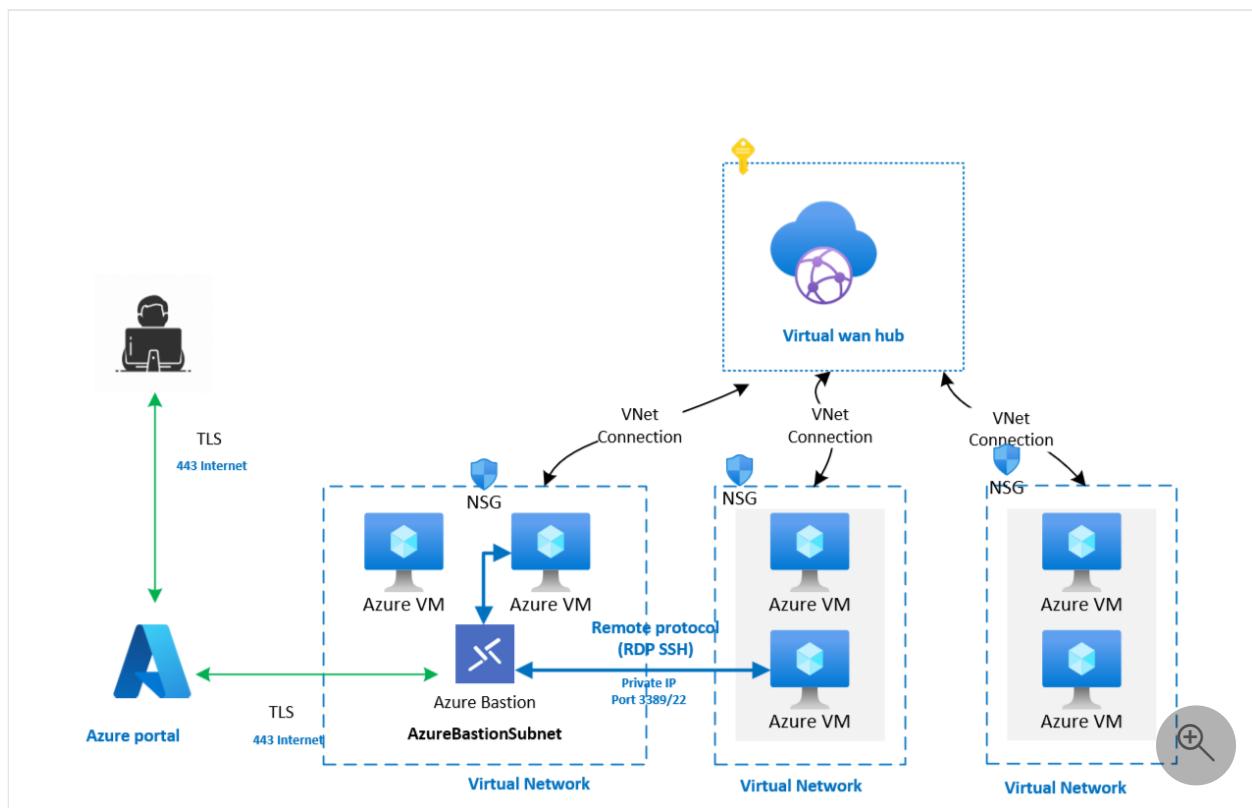


Figure 1: Azure Virtual WAN topology.

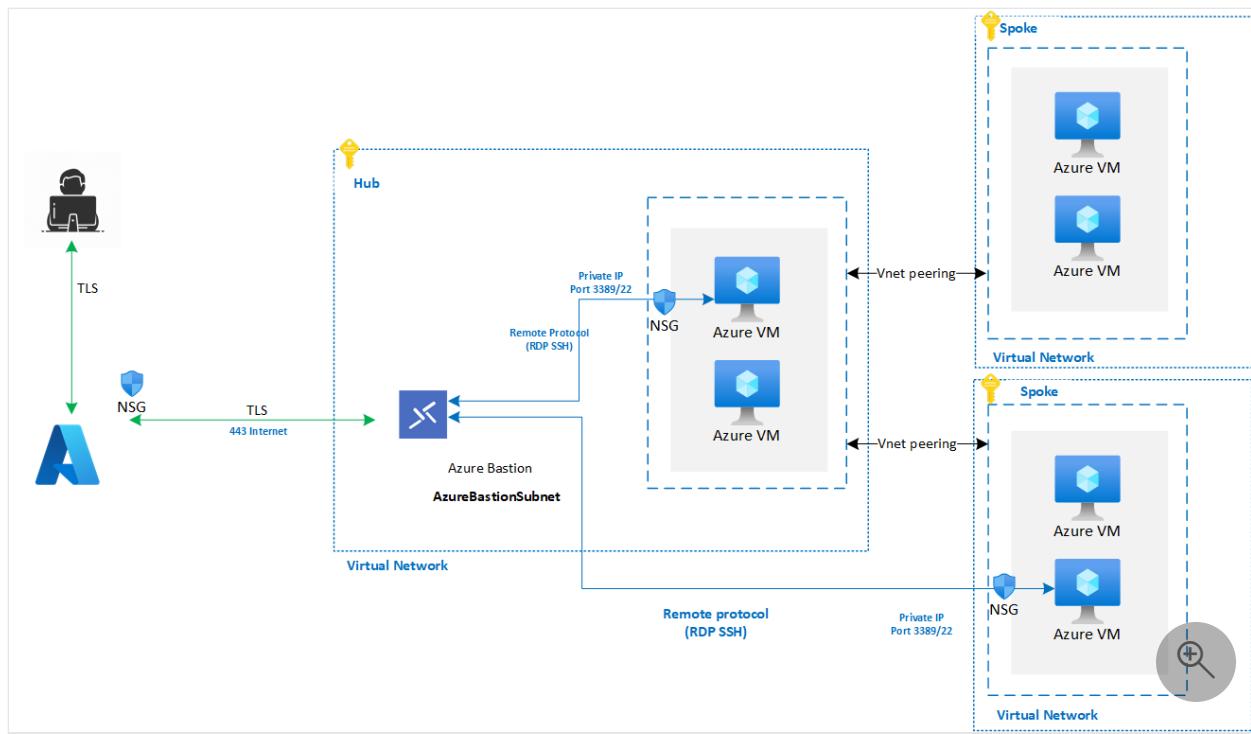


Figure 2: Azure hub-and-spoke topology.

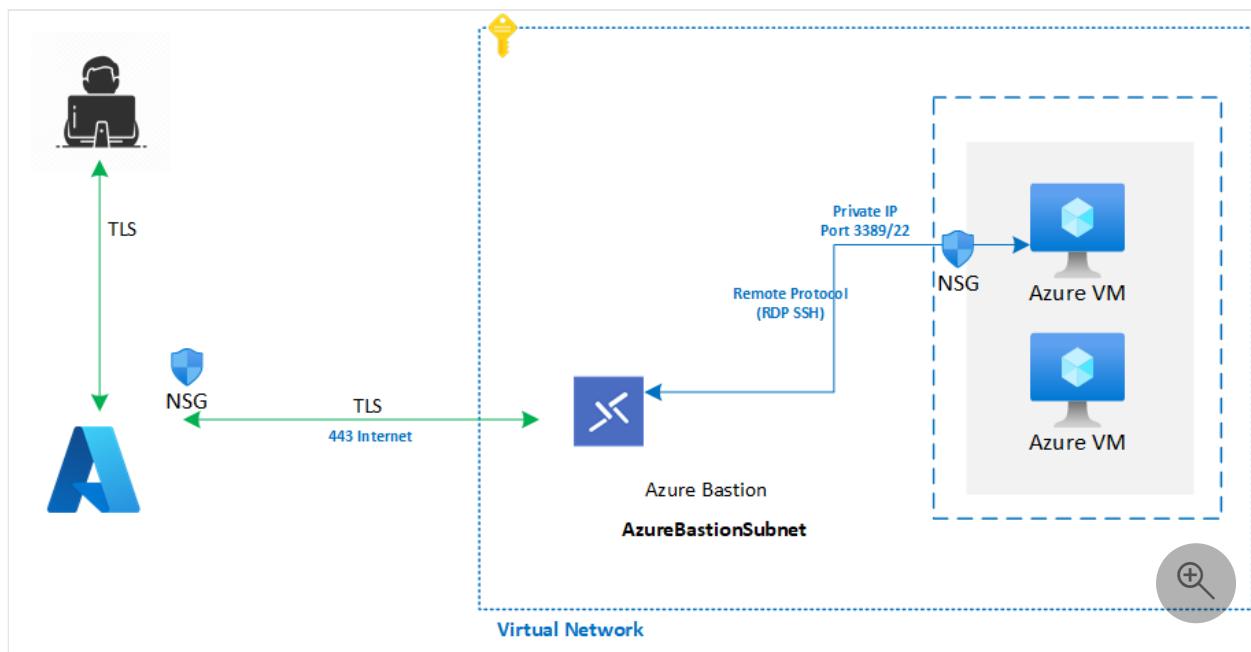


Figure 3: Azure standalone virtual network topology.

Azure Virtual Network Manager in Azure landing zones

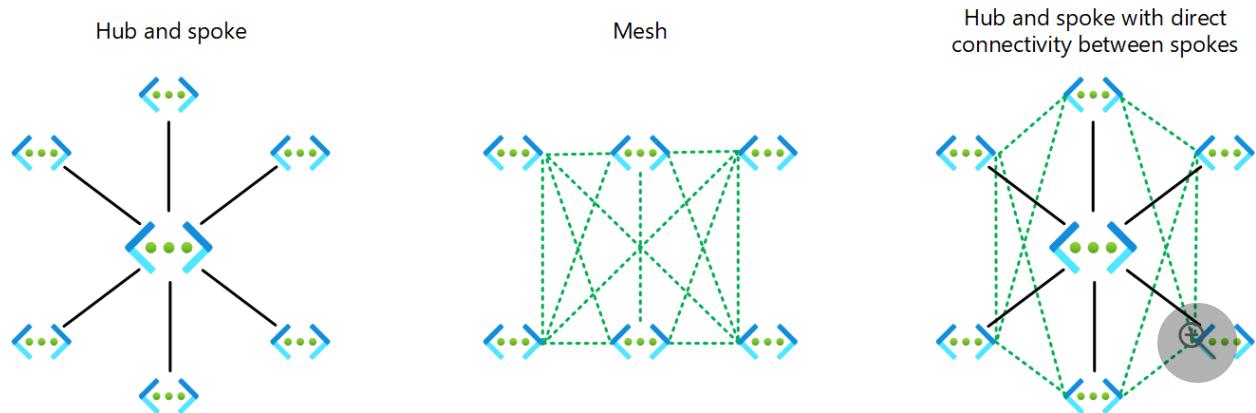
Article • 05/29/2024

This article describes how to use Virtual Network Manager to implement Azure landing zone design principles to accommodate application migrations, modernization, and innovation at scale. The Azure landing zones conceptual architecture recommends one of two networking topologies: a network topology that's based on Azure Virtual WAN or a network topology that's based on a traditional hub-and-spoke architecture.

You can use Virtual Network Manager to expand and implement networking changes as your business requirements change over time, such as if you require hybrid connectivity to migrate on-premises applications to Azure. In many cases, you can expand and implement networking changes without disrupting your deployed resources in Azure.

You can use Virtual Network Manager to create three types of [topologies](#) across subscriptions for both existing and new virtual networks:

- Hub-and-spoke topology
- Mesh topology (preview)
- Hub-and-spoke topology with direct connectivity between spokes



Virtual Network Manager supports Virtual WAN in Private Preview.

A hub-and-spoke topology with direct connectivity in Virtual Network Manager has spokes that connect to each other directly. The [connected group](#) feature automatically and bi-directionally enables direct connectivity between spoke virtual networks in the same [network group](#). Two connected groups can have the same virtual network.

You can use Virtual Network Manager to statically or dynamically add virtual networks to specific [network groups](#). Add virtual networks to specific network groups to define and

create your desired topology based on your connectivity configuration in Virtual Network Manager.

You can create multiple network groups to isolate groups of virtual networks from direct connectivity. Each network group provides the same region and multi-region support for spoke-to-spoke connectivity. Stay within the Virtual Network Manager-defined limits. For more information, see [Virtual Network Manager FAQ](#).

From a security perspective, Virtual Network Manager provides an efficient way to apply [security administrator rules](#) that deny or allow traffic flows centrally, regardless of the defined rules in the network security groups (NSGs). This capability allows network security administrators to enforce access controls and enable application owners to manage their own lower-level rules in NSGs.

You can use Virtual Network Manager to group virtual networks. You can then apply configurations to the groups rather than to individual virtual networks. Use this functionality to simultaneously manage connectivity, configuration and topology, security rules, and deployment to one or more regions without loss of fine-grained control.

You can segment networks by environments, teams, locations, lines of business, or some other function that suits your needs. To define network groups statically or dynamically, create a set of conditions that govern group membership.

Design considerations

- In a traditional hub-and-spoke deployment, you manually create and maintain virtual network peering connections. Virtual Network Manager introduces a layer of automation for virtual network peering, which makes large and complex network topologies like mesh easier to manage at scale. For more information, see [Network group overview](#).
- The security requirements of various business functions determine the need for creating network groups. A network group is a set of virtual networks that you select manually or through conditional statements. When you create a network group, you need to specify a policy, or Virtual Network Manager can create a policy if you explicitly allow it to. This policy enables Virtual Network Manager to be notified about changes. To update existing Azure policy initiatives, you need to deploy changes to the network group within the Virtual Network Manager resource.

- To design appropriate network groups, you should evaluate which parts of your network share common security characteristics. For example, you can create network groups for Corporate and Online to manage their connectivity and security rules at scale.
- When multiple virtual networks across your organization's subscriptions share the same security attributes, you can use Virtual Network Manager to apply them efficiently. You should, for example, place all systems that a business unit like HR or finance uses in a separate network group because you need to apply different admin rules to them.
- Virtual Network Manager can centrally apply security administrator rules, which have higher priority than NSG rules that are applied at the subnet level. (This feature is in preview.) This feature enables network and security teams to effectively enforce company policies and create security guardrails at scale. It also enables product teams to simultaneously maintain control of NSGs within their landing zone subscriptions.
- You can use the Virtual Network Manager [security administrator rules](#) feature to explicitly allow or deny specific network flows regardless of NSG configurations at the subnet or network interface levels. You can use this capability, for example, to always permit management services network flows. NSGs that application teams control can't override these rules.

Design recommendations

- Define the [scope of Virtual Network Manager](#). Apply security administrator rules that enforce organization-level rules at the root management group or the tenant. This strategy hierarchically applies rules automatically to existing resources, new resources, and to all associated management groups.
- Create a Virtual Network Manager instance in the Connectivity subscription with a [scope](#) of the intermediate root management group, such as Contoso. Enable the security administrator feature on this instance. This configuration allows you to define security administrator rules that apply across all virtual networks and subnets in your Azure landing zone hierarchy and helps you democratize NSGs to application landing zone owners and teams.
- Segment networks by grouping virtual networks either statically, which is a manual process, or dynamically, which is a policy-based process.

- Enable direct connectivity between spokes when selected spokes need to communicate frequently, with low latency and high throughput, and with each other, and when spokes need to access common services or network virtual appliances (NVAs) in the hub.
- Enable global mesh when all virtual networks across regions need to communicate with each other.
- Assign a priority value to each security administrator rule in your rule collections. The lower the value, the higher the priority of the rule.
- Use [security administrator rules](#) to explicitly allow or deny network flows, regardless of NSG configurations that application teams control. Use security administrator rules to fully delegate the control of NSGs and their rules to application teams.

Next step

[Automate management of user-defined routes \(UDRs\)](#)

Feedback

Was this page helpful?

 Yes

 No

Design area: Security

Article • 01/09/2024

This design area creates a foundation for security across your Azure, hybrid, and multicloud environments. You can enhance this foundation later with security guidance outlined in the [Secure methodology](#) of the Cloud Adoption Framework.

Design area review

Involved roles or functions: This design area is led by [cloud security](#), specifically the [security architects within that team](#). The [cloud platform](#) and [cloud center of excellence](#) are required to review networking and identity decisions. The collective roles might be required to define and implement the technical requirements coming from this exercise. More advanced security guardrails might also need support from [cloud governance](#).

Scope: The goal of this exercise is to understand security requirements and implement them consistently across all workloads in your cloud platform. The primary scope of this exercise focuses on security operations tooling and access control. This scope includes Zero Trust and advanced network security.

Out of scope: This exercise focuses on the foundation for a modern security operations center in the cloud. To streamline the conversation, this exercise doesn't address some of the disciplines in the [CAF Secure methodology](#). Security operations, asset protection, and innovation security will build on your Azure landing zone deployment. However, they're out of scope for this design area discussion.

Design area overview

Security is a core consideration for all customers, in every environment. When designing and implementing an Azure landing zone, security should be a consideration throughout the process.

The security design area focuses on considerations and recommendations for landing zone decisions. The [Secure methodology](#) of the Cloud Adoption Framework also provides further in-depth guidance for holistic security processes and tools.

New (greenfield) cloud environment: To start your cloud journey with a small set of subscriptions, see [Create your initial Azure subscriptions](#). Also, consider using Bicep deployment templates in building out your Azure landing zones. For more information, see [Azure Landing Zones Bicep - Deployment Flow](#).

Existing (brownfield) cloud environment: Consider using the following Microsoft Entra identity and access services if you are interested in applying the principles from security design area to existing Azure environments:

- Make use of Microsoft's [top 10 Azure security best practices](#). This guidance summarizes field-proven guidance from Microsoft cloud solution architects (CSAs) as well as Microsoft Partners.
- Deploy [Microsoft Entra Connect cloud sync](#) to provide your local Active Directory Domain Services (AD DS) users with secure single sign-on (SSO) to your Microsoft Entra ID-backed applications. An additional benefit to configuring hybrid identity is you can enforce [Microsoft Entra multifactor authentication \(MFA\)](#) and [Microsoft Entra Password Protection](#) to further protect these identities
- Consider [Microsoft Entra Conditional Access](#) to provided secure authentication to your cloud apps and Azure resources.
- Implement [Microsoft Entra Privileged Identity Management](#) to ensure least-privilege access and deep reporting in your entire Azure environment. Teams should begin recurring access reviews to ensure the right people and service principals have current and correct authorization levels.
- Make use of the recommendations, alerting, and remediation capabilities of [Microsoft Defender for Cloud](#). Your security team can also integrate Microsoft Defender for Cloud into [Microsoft Sentinel](#) if they need a more robust, centrally managed hybrid and multicloud Security Information Event Management (SIEM)/Security Orchestration and Response (SOAR) solution.

The [Azure Landing Zones Bicep - Deployment Flow](#) repository contains a number of Bicep deployment templates that can accelerate your greenfield and brownfield Azure landing zone deployments. These templates already have Microsoft proven-practice security guidance integrated within them.

For more information on working in brownfield cloud environments, see [Brownfield environment considerations](#).

Microsoft cloud security benchmark

The Microsoft cloud security benchmark includes high-impact security recommendations to help you secure most of the services you use in Azure. You can think of these recommendations as *general* or *organizational*, as they're applicable to most Azure services. The Microsoft cloud security benchmark recommendations are then customized for each Azure service. This customized guidance is contained in service recommendations articles.

The Microsoft cloud security benchmark documentation specifies security controls and service recommendations.

- **Security controls:** The Microsoft cloud security benchmark recommendations are categorized by security controls. Security controls represent high-level vendor-agnostic security requirements, like network security and data protection. Each security control has a set of security recommendations and instructions that help you implement those recommendations.
- **Service recommendations:** When available, benchmark recommendations for Azure services will include Microsoft cloud security benchmark recommendations that are tailored specifically for that service.

Azure Attestation

[Azure Attestation](#) is a tool that can help you ensure the security and integrity of your platform and binaries that run inside it. It's especially useful for businesses that require highly scalable compute resources and uncompromising trust with the remote attestation capability.

Security design considerations

An organization must have visibility into what's happening within their technical cloud estate. Security monitoring and audit logging of Azure platform services is a key component of a scalable framework.

Security operations design considerations

[] [Expand table](#)

Scope	Context
Security alerts	<ul style="list-style-type: none">- Which teams require notifications for security alerts?- Are there groups of services that alerts require routing to different teams?- Business requirements for real-time monitoring and alerting.- Security information and event management integration with Microsoft Defender for Cloud and Microsoft Sentinel.
Security logs	<ul style="list-style-type: none">- Data retention periods for audit data. Microsoft Entra ID P1 or P2 reports have a 30-day retention period.- Long-term archiving of logs like Azure activity logs, virtual machine (VM) logs, and platform as a service (PaaS) logs.

Scope	Context
Security controls	<ul style="list-style-type: none"> - Baseline security configuration via Azure in-guest VM policy. - Consider how your security controls will align with governance guardrails.
Vulnerability management	<ul style="list-style-type: none"> - Emergency patching for critical vulnerabilities. - Patching for VMs that are offline for extended periods of time. - Vulnerability assessment of VMs.
Shared responsibility	<ul style="list-style-type: none"> - Where are the handoffs for team responsibilities? These responsibilities need consideration when monitoring or responding to security events. - Consider the guidance in the Secure methodology for security operations.
Encryption and keys	<ul style="list-style-type: none"> - Who requires access to keys in the environment? - Who will be responsible for managing the keys? - Explore encryption and keys further.
Attestation	<ul style="list-style-type: none"> - Will you use Trusted Launch for your VMs, and do you need attestation of the integrity of the entire boot chain of your VM (UEFI, OS, system, and drivers)? - Do you want to take advantage of confidential disk encryption for your confidential VMs? - Do your workloads require attestation that they're running inside a trusted environment?

Security operations design recommendations

- Use [Microsoft Entra ID reporting capabilities](#) to generate access control audit reports.
- Export Azure activity logs to Azure Monitor Logs for long-term data retention. Export to Azure Storage for long-term storage beyond two years, if necessary.
- [Enable Defender for Cloud standard](#) for all subscriptions, and use Azure Policy to ensure compliance.
- Monitor base operating system patching drift via Azure Monitor Logs and Microsoft Defender for Cloud.
- Use Azure policies to automatically deploy software configurations through VM extensions and enforce a compliant baseline VM configuration.
- Monitor VM security configuration drift via Azure Policy.
- Connect default resource configurations to a centralized Azure Monitor Log Analytics workspace.

- Use an Azure Event Grid-based solution for log-oriented, real-time alerts.
- Use Azure Attestation for attestation of:
 - The integrity of the entire boot chain of your VM. For more information, see [Boot integrity monitoring overview](#).
 - Secure release of confidential disk encryption keys for a confidential VM. For more information, see [Confidential OS disk encryption](#).
 - Various types of workload trusted execution environments. For more information, see [Use cases](#).

Access control design considerations

Modern security boundaries are more complex than boundaries in a traditional datacenter. The four walls of the datacenter no longer contain your assets. Keeping users out of the protected network is no longer sufficient to control access. In the cloud, your perimeter is composed of two parts: network security controls and Zero Trust access controls.

Advanced network security

[] [Expand table](#)

Scope	Context
Plan for inbound and outbound internet connectivity	Describes recommended connectivity models for inbound and outbound connectivity to and from the public internet.
Plan for landing zone network segmentation	Explores key recommendations to deliver highly secure internal network segmentation within a landing zone. These recommendations drive network zero-trust implementation.
Define network encryption requirements	Explores key recommendations to achieve network encryption between on-premises and Azure and across Azure regions.
Plan for traffic inspection	Explores key considerations and recommended approaches for mirroring or tapping traffic within Azure Virtual Network.

Zero Trust

For Zero Trust access with identities, you should consider:

- Which teams or individuals require access to services within the landing zone?
What roles are they doing?

- Who should authorize the access requests?
- Who should receive the notifications when privileged roles are activated?
- Who should have access to the audit history?

For more information, see [Microsoft Entra Privileged Identity Management](#).

Implementing Zero Trust can go beyond just identity and access management. You should consider if your organization needs to implement Zero Trust practices across multiple pillars, such as infrastructure, data, and networking. For more information, see [Incorporate Zero Trust practices in your landing zone](#)

Access control design recommendations

- In the context of your underlying requirements, conduct a joint examination of each required service. If you want to bring your own keys, it might not be supported across all considered services. Implement relevant mitigation so that inconsistencies don't hinder wanted outcomes. Choose appropriate region pairs and disaster recovery regions that minimize latency.
- Develop a security allowlist plan to assess services like security configuration, monitoring, and alerts. Then create a plan to integrate them with existing systems.
- Determine the incident response plan for Azure services before moving it into production.
- Align your security requirements with Azure platform roadmaps to stay current with newly released security controls.
- Implement a [zero-trust approach for access](#) to the Azure platform where appropriate.

Security in the Azure landing zone accelerator

Security is at the core of the Azure landing zone accelerator. As part of the implementation, many tools and controls are deployed to help organizations quickly achieve a security baseline.

For example, the following are included:

Tools:

- Microsoft Defender for Cloud, standard or free tier
- Microsoft Sentinel

- Azure DDoS Network Protection (optional)
- Azure Firewall
- Web Application Firewall (WAF)
- Privileged Identity Management (PIM)

Policies for online and corporate-connected landing zones:

- Enforce secure access, like HTTPS, to storage accounts
- Enforce auditing for Azure SQL Database
- Enforce encryption for Azure SQL Database
- Prevent IP forwarding
- Prevent inbound RDP from internet
- Ensure subnets are associated with NSG

Next steps

Learn how to secure privileged access for hybrid and cloud deployments in Microsoft Entra ID.

[Secure privileged access](#)

Securing privileged access for hybrid and cloud deployments in Microsoft Entra ID

Article • 11/21/2024

Business asset security depends on the integrity of the privileged accounts that administer your IT systems. Cyber-attackers use credential theft attacks to target administrator accounts and other privileged access to try to gain access to sensitive data.

For cloud services, prevention and response are the joint responsibilities of the cloud service provider and the customer. For more information about the latest threats to endpoints and the cloud, see the [Microsoft Security Intelligence Report](#). This article can help you develop a roadmap toward closing the gaps between your current plans and the guidance described here.

Note

Microsoft is committed to the highest levels of trust, transparency, standards conformance, and regulatory compliance. Learn more about how the Microsoft global incident response team mitigates the effects of attacks against cloud services, and how security is built into Microsoft business products and cloud services at [Microsoft Trust Center - Security](#) and Microsoft compliance targets at [Microsoft Trust Center - Compliance](#).

Traditionally, organizational security was focused on the entry and exit points of a network as the security perimeter. However, SaaS apps and personal devices on the Internet have made this approach less effective.

In Microsoft Entra ID, we replace the network security perimeter with authentication in your organization's identity layer, with users assigned to privileged administrative roles in control. Their access must be protected, whether the environment is on-premises, cloud, or a hybrid.

Securing privileged access requires changes to:

- Processes, administrative practices, and knowledge management
- Technical components such as host defenses, account protections, and identity management

Secure your privileged access in a way that is managed and reported in the Microsoft services you care about. If you have on-premises administrator accounts, see the guidance for on-premises and hybrid privileged access in Active Directory at [Securing Privileged Access](#).

 **Note**

The guidance in this article refers primarily to features of Microsoft Entra ID that are included in Microsoft Entra ID P1 and P2. Microsoft Entra ID P2 is included in the EMS E5 suite and Microsoft 365 E5 suite. This guidance assumes your organization already has Microsoft Entra ID P2 licenses purchased for your users. If you do not have these licenses, some of the guidance might not apply to your organization. Also, throughout this article, the term Global Administrator means the same thing as "company administrator" or "tenant administrator."

Develop a roadmap

Microsoft recommends that you develop and follow a roadmap to secure privileged access against cyber attackers. You can always adjust your roadmap to accommodate your existing capabilities and specific requirements within your organization.

Each stage of the roadmap should raise the cost and difficulty for adversaries to attack privileged access for your on-premises, cloud, and hybrid assets. Microsoft recommends the following four roadmap stages. Schedule the most effective and the quickest implementations first.

This article can be your guide, based on Microsoft's experiences with cyber-attack incident and response implementation. The timelines for this roadmap are approximations.

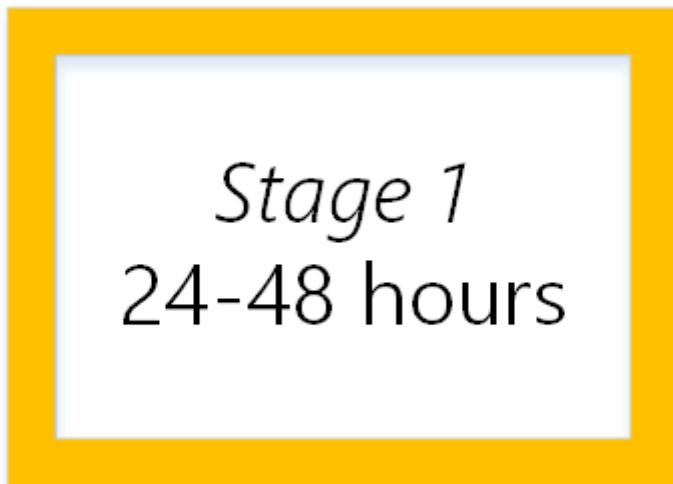


- Stage 1 (24-48 hours): Critical items that we recommend you do right away
- Stage 2 (2-4 weeks): Mitigate the most frequently used attack techniques
- Stage 3 (1-3 months): Build visibility and build full control of administrator activity

- Stage 4 (six months and beyond): Continue building defenses to further harden your security platform

This roadmap framework is designed to maximize the use of Microsoft technologies that you may have already deployed. Consider tying in to any security tools from other vendors that you have already deployed or are considering deploying.

Stage 1: Critical items to do right now



Stage 1 of the roadmap is focused on critical tasks that are fast and easy to implement. We recommend that you do these few items right away within the first 24-48 hours to ensure a basic level of secure privileged access. This stage of the Secured Privileged Access roadmap includes the following actions:

General preparation

Use Microsoft Entra Privileged Identity Management

We recommend that you start using Microsoft Entra Privileged Identity Management (PIM) in your Microsoft Entra production environment. After you start using PIM, you'll receive notification email messages for privileged access role changes. Notifications provide early warning when additional users are added to highly privileged roles.

Microsoft Entra Privileged Identity Management is included in Microsoft Entra ID P2 or EMS E5. To help you protect access to applications and resources on-premises and in the cloud, sign up for the [Enterprise Mobility + Security free 90-day trial](#). Microsoft Entra Privileged Identity Management and Microsoft Entra ID Protection monitor security activity using Microsoft Entra ID reporting, auditing, and alerts.

After you start using Microsoft Entra Privileged Identity Management:

1. Sign in to the Microsoft Entra admin center [↗](#) as a **Global Administrator**.
2. To switch directories where you want to use Privileged Identity Management, select your user name in the upper right corner of the Microsoft Entra admin center.
3. Browse to **Identity governance > Privileged Identity Management**.

Make sure the first person to use PIM in your organization is assigned to the **Security Administrator** and **Privileged Role Administrator** roles. Only Privileged Role Administrators can manage the Microsoft Entra directory role assignments of users. The PIM security wizard walks you through the initial discovery and assignment experience. You can exit the wizard without making any additional changes at this time.

Identify and categorize accounts that are in highly privileged roles

After starting to use Microsoft Entra Privileged Identity Management, view the users who are in the following Microsoft Entra roles:

- Global Administrator
- Privileged Role Administrator
- Exchange Administrator
- SharePoint Administrator

If you don't have Microsoft Entra Privileged Identity Management in your organization, you can use [Microsoft Graph PowerShell](#). Start with the Global Administrator role because a Global Administrator has the same permissions across all cloud services for which your organization has subscribed. These permissions are granted no matter where they were assigned: in the Microsoft 365 admin center, the Microsoft Entra admin center, or by using Microsoft Graph PowerShell.

Remove any accounts that are no longer needed in those roles. Then, categorize the remaining accounts that are assigned to administrator roles:

- Assigned to administrative users, but also used for non-administrative purposes (for example, personal email)
- Assigned to administrative users and used for administrative purposes only
- Shared across multiple users
- For break-glass emergency access scenarios
- For automated scripts
- For external users

Define at least two emergency access accounts

Microsoft recommends that organizations have two cloud-only emergency access accounts permanently assigned the [Global Administrator](#) role. These accounts are highly privileged and aren't assigned to specific individuals. The accounts are limited to emergency or "break glass" scenarios where normal accounts can't be used or all other administrators are accidentally locked out. These accounts should be created following the [emergency access account recommendations](#).

Turn on multifactor authentication and register all other highly privileged single-user non-federated administrator accounts

Require Microsoft Entra multifactor authentication at sign-in for all individual users who are permanently assigned to one or more of the Microsoft Entra administrator roles: Global Administrator, Privileged Role Administrator, Exchange Administrator, and SharePoint Administrator. Use the guidance at [Enforce multifactor authentication on your administrators](#) and ensure that all those users have registered at <https://aka.ms/mfasetup>. More information can be found under step 2 and step 3 of the guide [Protect user and device access in Microsoft 365](#).

Stage 2: Mitigate frequently used attacks



Stage 2
2-4 weeks

Stage 2 of the roadmap focuses on mitigating the most frequently used attack techniques of credential theft and abuse and can be implemented in approximately 2-4 weeks. This stage of the Secured Privileged Access roadmap includes the following actions.

General preparation

Conduct an inventory of services, owners, and administrators

The increase in "bring your own device" and work from home policies and the growth of wireless connectivity make it critical to monitor who is connecting to your network. A security audit can reveal devices, applications, and programs on your network that your organization doesn't support and that represent high risk. For more information, see [Azure security management and monitoring overview](#). Ensure that you include all of the following tasks in your inventory process.

- Identify the users who have administrative roles and the services where they can manage.
- Use Microsoft Entra PIM to find out which users in your organization have administrator access to Microsoft Entra ID.
- Beyond the roles defined in Microsoft Entra ID, Microsoft 365 comes with a set of administrator roles that you can assign to users in your organization. Each administrator role maps to common business functions, and gives people in your organization permissions to do specific tasks in the [Microsoft 365 admin center](#). Use the Microsoft 365 admin center to find out which users in your organization have administrator access to Microsoft 365, including via roles not managed in Microsoft Entra ID. For more information, see [About Microsoft 365 administrator roles](#) and [Security practices for Office 365](#).
- Do the inventory in services your organization relies on, such as Azure, Intune, or Dynamics 365.
- Ensure that your accounts that are used for administration purposes:
 - Have working email addresses attached to them
 - Have registered for Microsoft Entra multifactor authentication or use MFA on-premises
- Ask users for their business justification for administrative access.
- Remove administrator access for those individuals and services that don't need it.

Identify Microsoft accounts in administrative roles that need to be switched to work or school accounts

If your initial Global Administrators reuse their existing Microsoft account credentials when they began using Microsoft Entra ID, replace the Microsoft accounts with individual cloud-based or synchronized accounts.

Ensure separate user accounts and mail forwarding for Global Administrator accounts

Personal email accounts are regularly phished by cyber attackers, a risk that makes personal email addresses unacceptable for Global Administrator accounts. To help separate internet risks from administrative privileges, create dedicated accounts for each user with administrative privileges.

- Be sure to create separate accounts for users to do Global Administrator tasks.
- Make sure that your Global Administrators don't accidentally open emails or run programs with their administrator accounts.
- Be sure those accounts have their email forwarded to a working mailbox.
- Global Administrator (and other privileged groups) accounts should be cloud-only accounts with no ties to on-premises Active Directory.

Ensure the passwords of administrative accounts have recently changed

Ensure all users have signed into their administrative accounts and changed their passwords at least once in the last 90 days. Also, verify that any shared accounts have had their passwords changed recently.

Turn on password hash synchronization

Microsoft Entra Connect synchronizes a hash of the hash of a user's password from on-premises Active Directory to a cloud-based Microsoft Entra organization. You can use password hash synchronization as a backup if you use federation with Active Directory Federation Services (AD FS). This backup can be useful if your on-premises Active Directory or AD FS servers are temporarily unavailable.

Password hash sync enables users to sign in to a service by using the same password they use to sign in to their on-premises Active Directory instance. Password hash sync allows Microsoft Entra ID Protection to detect compromised credentials by comparing password hashes with passwords known to be compromised. For more information, see [Implement password hash synchronization with Microsoft Entra Connect Sync](#).

Require multifactor authentication for users in privileged roles and exposed users

Microsoft Entra ID recommends that you require multifactor authentication for all of your users. Be sure to consider users who would have a significant impact if their account were compromised (for example, financial officers). MFA reduces the risk of an attack because of a compromised password.

Turn on:

- MFA using Conditional Access policies for all users in your organization.

If you use Windows Hello for Business, the MFA requirement can be met using the Windows Hello sign-in experience. For more information, see [Windows Hello](#).

Microsoft Entra ID Protection

Microsoft Entra ID Protection is an algorithm-based monitoring and reporting tool that detects potential vulnerabilities affecting your organization's identities. You can configure automated responses to those detected suspicious activities, and take appropriate action to resolve them. For more information, see [Microsoft Entra ID Protection](#).

Obtain your Microsoft 365 Secure Score (if using Microsoft 365)

Secure Score looks at your settings and activities for the Microsoft 365 services you're using and compares them to a baseline established by Microsoft. You'll get a score based on how aligned you are with security practices. Anyone who has the administrator permissions for a Microsoft 365 Business Standard or Enterprise subscription can access the Secure Score at <https://security.microsoft.com/securescore>.

Review the Microsoft 365 security and compliance guidance (if using Microsoft 365)

The [plan for security and compliance](#) outlines the approach for an Office 365 customer to configure Office 365 and enable other EMS capabilities. Then, review steps 3-6 of how to [Protect access to data and services in Microsoft 365](#) and the guide for how to [monitor security and compliance in Microsoft 365](#).

Configure Microsoft 365 Activity Monitoring (if using Microsoft 365)

Monitor your organization for users who are using Microsoft 365 to identify staff who have an administrator account but might not need Microsoft 365 access because they don't sign in to those portals. For more information, see [Activity reports in the Microsoft 365 admin center](#).

Establish incident/emergency response plan owners

Establishing a successful incident response capability requires considerable planning and resources. You must continually monitor for cyber-attacks and establish priorities for incident handling. Collect, analyze, and report incident data to build relationships and establish communication with other internal groups and plan owners. For more information, see [Microsoft Security Response Center](#).

Secure on-premises privileged administrative accounts, if not already done

If your Microsoft Entra organization is synchronized with on-premises Active Directory, then follow the guidance in [Security Privileged Access Roadmap](#): This stage includes:

- Creating separate administrator accounts for users who need to conduct on-premises administrative tasks
- Deploying Privileged Access Workstations for Active Directory administrators
- Creating unique local administrator passwords for workstations and servers

Additional steps for organizations managing access to Azure

Complete an inventory of subscriptions

Use the Enterprise portal and the Azure portal to identify the subscriptions in your organization that host production applications.

Remove Microsoft accounts from administrator roles

Microsoft accounts from other programs, such as Xbox, Live, and Outlook, shouldn't be used as administrator accounts for your organization's subscriptions. Remove administrator status from all Microsoft accounts, and replace with Microsoft Entra ID (for example, chris@contoso.com) work or school accounts. For administrator purposes, depend on accounts that are authenticated in Microsoft Entra ID and not in other services.

Monitor Azure activity

The Azure Activity Log provides a history of subscription-level events in Azure. It offers information about who created, updated, and deleted what resources, and when these events occurred. For more information, see [Audit and receive notifications about important actions in your Azure subscription](#).

Additional steps for organizations managing access to other cloud apps via Microsoft Entra ID

Configure Conditional Access policies

Prepare Conditional Access policies for on-premises and cloud-hosted applications. If you have users workplace joined devices, get more information from [Setting up on-premises Conditional Access by using Microsoft Entra device registration](#).

Stage 3: Take control of administrator activity



Stage 3 builds on the mitigations from Stage 2 and should be implemented in approximately 1-3 months. This stage of the Secured Privileged Access roadmap includes the following components.

General preparation

Complete an access review of users in administrator roles

More corporate users are gaining privileged access through cloud services, which can lead to un-managed access. Users today can become Global Administrators for Microsoft 365, Azure subscription administrators, or have administrator access to VMs or via SaaS apps.

Your organization should have all employees handle ordinary business transactions as unprivileged users, and then grant administrator rights only as needed. Complete access reviews to identify and confirm the users who are eligible to activate administrator privileges.

We recommend that you:

1. Determine which users are Microsoft Entra administrators, enable on-demand, just-in-time administrator access, and role-based security controls.
2. Convert users who have no clear justification for administrator privileged access to a different role (if no eligible role, remove them).

Continue rollout of stronger authentication for all users

Require highly exposed users to have modern, strong authentication such as Microsoft Entra multifactor authentication or Windows Hello. Examples of highly exposed users include:

- C-suite executives
- High-level managers
- Critical IT and security personnel

Use dedicated workstations for administration for Microsoft Entra ID

Attackers might try to target privileged accounts so that they can disrupt the integrity and authenticity of data. They often use malicious code that alters the program logic or snoops the administrator entering a credential. Privileged Access Workstations (PAWs) provide a dedicated operating system for sensitive tasks that is protected from Internet attacks and threat vectors. Separating these sensitive tasks and accounts from the daily use workstations and devices provides strong protection from:

- Phishing attacks
- Application and operating system vulnerabilities
- Impersonation attacks
- Credential theft attacks such as keystroke logging, Pass-the-Hash, and Pass-The-Ticket

By deploying privileged access workstations, you can reduce the risk that administrators enter their credentials in a desktop environment that hasn't been hardened. For more information, see [Privileged Access Workstations](#).

Review National Institute of Standards and Technology recommendations for handling incidents

The National Institute of Standards and Technology's (NIST) provides guidelines for incident handling, particularly for analyzing incident-related data and determining the

appropriate response to each incident. For more information, see [The \(NIST\) Computer Security Incident Handling Guide \(SP 800-61, Revision 2\)](#).

Implement Privileged Identity Management (PIM) for JIT to additional administrative roles

For Microsoft Entra ID, use [Microsoft Entra Privileged Identity Management](#) capability. Time-limited activation of privileged roles works by enabling you to:

- Activate administrator privileges to do a specific task
- Enforce MFA during the activation process
- Use alerts to inform administrators about out-of-band changes
- Enable users to keep their privileged access for a pre-configured amount of time
- Allow security administrators to:
 - Discover all privileged identities
 - View audit reports
 - Create access reviews to identify every user who is eligible to activate administrator privileges

If you're already using Microsoft Entra Privileged Identity Management, adjust timeframes for time-bound privileges as necessary (for example, maintenance windows).

Determine exposure to password-based sign-in protocols (if using Exchange Online)

We recommend you identify every potential user who could be catastrophic to the organization if their credentials were compromised. For those users, put in place strong authentication requirements and use Microsoft Entra Conditional Access to keep them from signing in to their email using username and password. You can block [legacy authentication using Conditional Access](#), and you can [block basic authentication](#) through Exchange Online.

Complete a roles review assessment for Microsoft 365 roles (if using Microsoft 365)

Assess whether all administrators users are in the correct roles (delete and reassign according to this assessment).

Review the security incident management approach used in Microsoft 365 and compare with your own organization

You can download this report from [Security Incident Management in Microsoft 365](#).

Continue to secure on-premises privileged administrative accounts

If your Microsoft Entra ID is connected to on-premises Active Directory, then follow the guidance in the [Security Privileged Access Roadmap](#): Stage 2. In this stage, you:

- Deploy Privileged Access Workstations for all administrators
- Require MFA
- Use Just Enough Admin for domain controller maintenance, lowering the attack surface of domains
- Deploy [Advanced Threat Analytics](#) for attack detection

Additional steps for organizations managing access to Azure

Establish integrated monitoring

The [Microsoft Defender for Cloud](#):

- Provides integrated security monitoring and policy management across your Azure subscriptions
- Helps detect threats that may otherwise go unnoticed
- Works with a broad array of security solutions

Inventory your privileged accounts within hosted Virtual Machines

You don't usually need to give users unrestricted permissions to all your Azure subscriptions or resources. Use Microsoft Entra administrator roles to grant only the access that your users who need to do their jobs. You can use Microsoft Entra administrator roles to let one administrator manage only VMs in a subscription, while another can manage SQL databases within the same subscription. For more information, see [What is Azure role-based access control](#).

Implement PIM for Microsoft Entra administrator roles

Use Privileged identity Management with Microsoft Entra administrator roles to manage, control, and monitor access to Azure resources. Using PIM protects by lowering the

exposure time of privileges and increasing your visibility into their use through reports and alerts. For more information, see [What is Microsoft Entra Privileged Identity Management](#).

Use Azure log integrations to send relevant Azure logs to your SIEM systems

Azure log integration enables you to integrate raw logs from your Azure resources to your organization's existing Security Information and Event Management (SIEM) systems. [Azure log integration](#) collects Windows events from Windows Event Viewer logs and Azure resources from:

- Azure activity Logs
- Microsoft Defender for Cloud alerts
- Azure resource logs

Additional steps for organizations managing access to other cloud apps via Microsoft Entra ID

Implement user provisioning for connected apps

Microsoft Entra ID allows you to automate creating and maintaining user identities in cloud apps like Dropbox, Salesforce, and ServiceNow. For more information, see [Automate user provisioning and deprovisioning to SaaS applications with Microsoft Entra ID](#).

Integrate information protection

Microsoft Defender for Cloud Apps allows you to investigate files and set policies based on Azure Information Protection classification labels, enabling greater visibility and control of your cloud data. Scan and classify files in the cloud and apply Azure information protection labels. For more information, see [Azure Information Protection integration](#).

Configure Conditional Access

Configure Conditional Access based on a group, location, and application sensitivity for [SaaS apps](#) and Microsoft Entra connected apps.

Monitor activity in connected cloud apps

We recommend using [Microsoft Defender for Cloud Apps](#) to ensure that user access is also protected in connected applications. This feature secures the enterprise access to cloud apps and secures your administrator accounts, allowing you to:

- Extend visibility and control to cloud apps
- Create policies for access, activities, and data sharing
- Automatically identify risky activities, abnormal behaviors, and threats
- Prevent data leakage
- Minimize risk and automated threat prevention and policy enforcement

The Defender for Cloud Apps SIEM agent integrates Defender for Cloud Apps with your SIEM server to enable centralized monitoring of Microsoft 365 alerts and activities. It runs on your server and pulls alerts and activities from Defender for Cloud Apps and streams them into the SIEM server. For more information, see [SIEM integration](#).

Stage 4: Continue building defenses



Stage 4
6 months and
beyond

Stage 4 of the roadmap should be implemented at six months and beyond. Complete your roadmap to strengthen your privileged access protections from potential attacks that are known today. For the security threats of tomorrow, we recommend viewing security as an ongoing process to raise the costs and reduce the success rate of adversaries targeting your environment.

Securing privileged access is important to establish security assurances for your business assets. However, it should be part of a complete security program that provides ongoing security assurances. This program should include elements such as:

- Policy
- Operations
- Information security
- Servers

- Applications
- PCs
- Devices
- Cloud fabric

We recommend the following practices when you're managing privileged access accounts:

- Ensure that administrators are doing their day-to-day business as unprivileged users
- Grant privileged access only when needed, and remove it afterward (just-in-time)
- Keep audit activity logs relating to privileged accounts

For more information on building a complete security roadmap, see [Microsoft cloud IT architecture resources](#). To engage with Microsoft services to help you implement any part of your roadmap, contact your Microsoft representative or see [Build critical cyber defenses to protect your enterprise](#).

This final ongoing stage of the Secured Privileged Access roadmap includes the following components.

General preparation

Review administrator roles in Microsoft Entra ID

Determine if current built-in Microsoft Entra administrator roles are still up to date and ensure that users are in only the roles they need. With Microsoft Entra ID, you can assign separate administrators to serve different functions. For more information, see [Microsoft Entra built-in roles](#).

Review users who have administration of Microsoft Entra joined devices

For more information, see [How to configure Microsoft Entra hybrid joined devices](#).

Review members of [built-in Microsoft 365 admin roles](#)

Skip this step if you're not using Microsoft 365.

Validate incident response plan

To improve upon your plan, Microsoft recommends you regularly validate that your plan operates as expected:

- Go through your existing road map to see what was missed
- Based on the postmortem analysis, revise existing or define new practices
- Ensure that your updated incident response plan and practices are distributed throughout your organization

Additional steps for organizations managing access to Azure

Determine if you need to [transfer ownership of an Azure subscription to another account](#).

"Break glass": what to do in an emergency



1. Notify key managers and security officers with information about the incident.
2. Review your attack playbook.
3. Access your "break glass" account username and password combination to sign in to Microsoft Entra ID.
4. Get help from Microsoft by [opening an Azure support request](#).
5. Look at the [Microsoft Entra sign-in reports](#). There might be some time between an event occurring and when it's included in the report.
6. For hybrid environments, if your on-premises infrastructure federated and your AD FS server aren't available, you can temporarily switch from federated authentication to use password hash sync. This switch reverts the domain federation back to managed authentication until the AD FS server becomes available.
7. Monitor email for privileged accounts.

8. Make sure you save backups of relevant logs for potential forensic and legal investigation.

For more information about how Microsoft Office 365 handles security incidents, see [Security Incident Management in Microsoft Office 365](#).

FAQ: Answers for securing privileged access

Q: What do I do if I haven't implemented any secure access components yet?

Answer: Define at least two break-glass account, assign MFA to your privileged administrator accounts, and separate user accounts from Global Administrator accounts.

Q: After a breach, what is the top issue that needs to be addressed first?

Answer: Be sure you're requiring the strongest authentication for highly exposed individuals.

Q: What happens if our privileged administrators have been deactivated?

Answer: Create a Global Administrator account that is always kept up to date.

Q: What happens if there's only one Global Administrator left and they can't be reached?

Answer: Use one of your break-glass accounts to gain immediate privileged access.

Q: How can I protect administrators within my organization?

Answer: Have administrators always do their day-to-day business as standard "unprivileged" users.

Q: What are the best practices for creating administrator accounts within Microsoft Entra ID?

Answer: Reserve privileged access for specific administrator tasks.

Q: What tools exist for reducing persistent administrator access?

Answer: Privileged Identity Management (PIM) and Microsoft Entra administrator roles.

Q: What is the Microsoft position on synchronizing administrator accounts to Microsoft Entra ID?

Answer: Tier 0 administrator accounts are used only for on-premises AD accounts. Such accounts aren't typically synchronized with Microsoft Entra ID in the cloud. Tier 0

administrator accounts include accounts, groups, and other assets that have direct or indirect administrative control of the on-premises Active Directory forest, domains, domain controllers, and assets.

Q: How do we keep administrators from assigning random administrator access in the portal?

Answer: Use non-privileged accounts for all users and most administrators. Start by developing a footprint of the organization to determine which few administrator accounts should be privileged. And monitor for newly created administrative users.

Next steps

- [Microsoft Trust Center for Product Security](#) – Security features of Microsoft cloud products and services
- [Microsoft compliance offerings](#) – Microsoft's comprehensive set of compliance offerings for cloud services
- [Guidance on how to do a risk assessment](#) – Manage security and compliance requirements for Microsoft cloud services

Other Microsoft Online Services

- [Microsoft Intune Security](#) – Intune provides mobile device management, mobile application management, and PC management capabilities from the cloud.
- [Microsoft Dynamics 365 security](#) – Dynamics 365 is the Microsoft cloud-based solution that unifies customer relationship management (CRM) and enterprise resource planning (ERP) capabilities.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#)

Encryption and key management in Azure

Article • 08/15/2023

Encryption is a vital step toward ensuring data privacy, compliance, and data residency in Microsoft Azure. It's also one of the most important security concerns of many enterprises. This section covers design considerations and recommendations for encryption and key management.

Design considerations

- Set subscription and scale limits as they apply to Azure Key Vault.

Key Vault has transaction limits for keys and secrets. To [throttle transactions](#) per vault for a certain period of time, see [Azure limits](#).

Key Vault serves a security boundary because access permissions for keys, secrets, and certificates are at the vault level. Key Vault access policy assignments grant permissions separately to keys, secrets, or certificates. They don't support granular, object-level permissions like a specific key, secret, or certificate [key management](#).

- Isolate application-specific and workload-specific secrets and shared secrets, as necessary, to [control access](#).
- Optimize Premium SKUs where HSM-protected (Hardware Security Module) keys are required.

Underlying HSMs are FIPS 140-2 Level 2 compliant. Manage Azure dedicated HSM for FIPS 140-2 Level 3 compliance by considering the supported scenarios.

- Manage key rotation and secret expiration.
- Use [Key Vault certificates](#) to manage certificate procurement and signing. Set alerting, notifications, and automated certificate renewals.
- Set disaster recovery requirements for keys, certificates, and secrets.
- Set Key Vault service replication and failover capabilities. Set [availability and redundancy](#).
- Monitor key, certificate, and secret usage.

Detect unauthorized access by using a key vault or Azure Monitor Log Analytics workspace. For more information, see [Monitoring and alerting for Azure Key Vault](#).

- Delegate Key Vault instantiation and privileged access. For more information, see [Azure Key Vault security](#).
- Set requirements for using customer-managed keys for native encryption mechanisms, such as Azure Storage encryption:
 - [Customer-managed keys](#)
 - Whole-disk encryption for virtual machines (VMs)
 - Data-in-transit encryption
 - Data-at-rest encryption

Design recommendations

- Use a federated Azure Key Vault model to avoid transaction scale limits.
- Azure RBAC is the recommended authorization system for the Azure Key Vault data plane. See [Azure role-based access control \(Azure RBAC\) vs. access policies \(legacy\)](#) for more information.
- Provision Azure Key Vault with the soft delete and purge policies enabled to allow retention protection for deleted objects.
- Follow a least-privilege model by limiting the authorization to permanently delete keys, secrets, and certificates to specialized custom Azure Active Directory (Azure AD) roles.
- Automate the certificate management and renewal process with public certificate authorities to ease administration.
- Establish an automated process for key and certificate rotation.
- Enable firewall and virtual network service endpoints on the vault to control access to the key vault.
- Use the platform-central Azure Monitor Log Analytics workspace to audit key, certificate, and secret usage within each instance of Key Vault.
- Delegate Key Vault instantiation and privileged access, and use Azure Policy to enforce a consistent compliant configuration.
- Default to Microsoft-managed keys for principal encryption functionality, and use customer-managed keys when required.

- Don't use centralized instances of Key Vault for application keys or secrets.
- To avoid secret sharing across environments, don't share Key Vault instances between applications.

Service enablement framework

Article • 10/09/2023

As business units request to deploy workloads to Azure, you need visibility into each workload to determine how to achieve the right governance, security, and compliance levels. When a new service is required, you need to allow it.

The following tables provide a framework to assess the enterprise security readiness of Azure services.

Security

Category	Criteria
Network endpoint	<ul style="list-style-type: none">- Does the service have a public endpoint accessible outside of a virtual network?- Does it support virtual network service endpoints?- Can Azure services interact directly with the service endpoint?- Does it support Azure Private Link endpoints?- Can it be deployed within a virtual network?
Data exfiltration prevention	<ul style="list-style-type: none">- Does the Platform-as-a-Service (PaaS) service have a separate Border Gateway Protocol (BGP) community in Azure ExpressRoute Microsoft peering?- Does ExpressRoute expose a route filter for the service?- Does the service support Private Link endpoints?
Enforce network traffic flow for management and data plane operations	<ul style="list-style-type: none">- Is it possible to inspect traffic entering and exiting the service?- Can traffic be force-tunneled with user-defined routing?- Do management operations use Azure shared public IP ranges?- Is management traffic directed via a link-local endpoint exposed on the host?
Data encryption at-rest	<ul style="list-style-type: none">- Is encryption applied by default?- Can encryption be disabled?- Is encryption done with Microsoft-managed keys or customer-managed keys?
Data encryption in-transit	<ul style="list-style-type: none">- Is traffic to the service encrypted at a protocol level, like SSL/TLS?- Are there any HTTP endpoints, and can they be disabled?- Is the underlying service communication encrypted?- Is encryption done with Microsoft-managed keys or

Category	Criteria
	customer-managed keys? Is bringing your own encryption supported?
Software deployment	<ul style="list-style-type: none"> - Can application software or third-party products be deployed to the service? - How is software deployment done and managed? - Can policies be enforced to control source code integrity? - Can antimalware capability, vulnerability management, and security monitoring tools be used if the software is deployable? - Does the service provide such capabilities natively, such as with Azure Kubernetes Service (AKS)?

Identity and access management

Category	Criteria
Authentication and access control	<ul style="list-style-type: none"> - Are all control plane operations governed by Microsoft Entra ID? Is there a nested control plane, such as with AKS? - What methods exist to provide access to the data plane? - Does the data plane integrate with Microsoft Entra ID? - Does authentication between Azure services use managed identities or service principals? - How are any applicable keys or shared access signatures managed? - How can access be revoked?
Segregation of duties	Does the service separate control plane and data plane operations within Microsoft Entra ID?
Multifactor authentication and conditional access	Is multifactor authentication enforced for user-to-service interactions?

Governance

Category	Criteria
Data export and import	Can you import and export data securely and encrypted with the service?
Data privacy and usage	<ul style="list-style-type: none"> - Can Microsoft engineers access the data? - Is any Microsoft Support interaction with the service audited?
Data residency	Is data contained in the service deployment region?

Operations

Category	Criteria
Monitoring	Does the service integrate with Azure Monitor?
Backup management	<ul style="list-style-type: none">- Which workload data needs to be backed up?- How are backups captured?- How frequently can backups be taken?- How long can backups be kept for?- Are backups encrypted?- Is backup encryption done with Microsoft-managed keys or customer-managed keys?
Disaster recovery	<ul style="list-style-type: none">- How can the service be used in a regionally redundant fashion?- What are the achievable recovery time and recovery point goals?
SKU	<ul style="list-style-type: none">- What SKUs are available? How do they differ?- Are there any features related to security for the Premium SKU?
Capacity management	<ul style="list-style-type: none">- How is capacity monitored?- What is the unit of horizontal scale?
Patch and update management	<ul style="list-style-type: none">- Does the service require active updating, or do updates happen automatically?- How frequently are updates applied? Can they be automated?
Audit	<ul style="list-style-type: none">- Are nested control plane operations captured? For example, AKS or Azure Databricks.- Are key data plane activities recorded?
Configuration management	Does it support tags and provide a <code>put</code> schema for all resources?

Azure service compliance

Category	Criteria
Service attestation, certification, and external audits	Is the service PCI/ISO/SOC compliant?
Service availability	<ul style="list-style-type: none">- Is the service generally available?- In what regions is the service available?- What is the deployment scope of the service? Is it a regional or global service?
Service-level agreements (SLAs)	<ul style="list-style-type: none">- What is the SLA for service availability?- If applicable, what is the SLA for performance?

Security control mapping with Azure landing zones

Article • 12/01/2022

Many organizations are required to comply with certain industry/regional regulations before adopting and onboarding the Azure cloud services. These compliance regulations are identified by compliance domain and controls respectively. For example, **CMMC L3 AC 1.001** where AC is Access Control domain and 1.001 is a control ID in Cybersecurity Maturity Model Certification (CMMC) framework. The best practice recommendation is to map the required compliance controls to Microsoft cloud security benchmark (MCSB) and identify the custom set of the controls that are not covered by MCSB.

In Addition, MCSB also provides the list of built-in policies and Policy initiatives GUIDs to addresses the required controls. For the controls that are not covered in MCSB, the control mapping guidance includes a step by step process on how to build policies and initiatives.

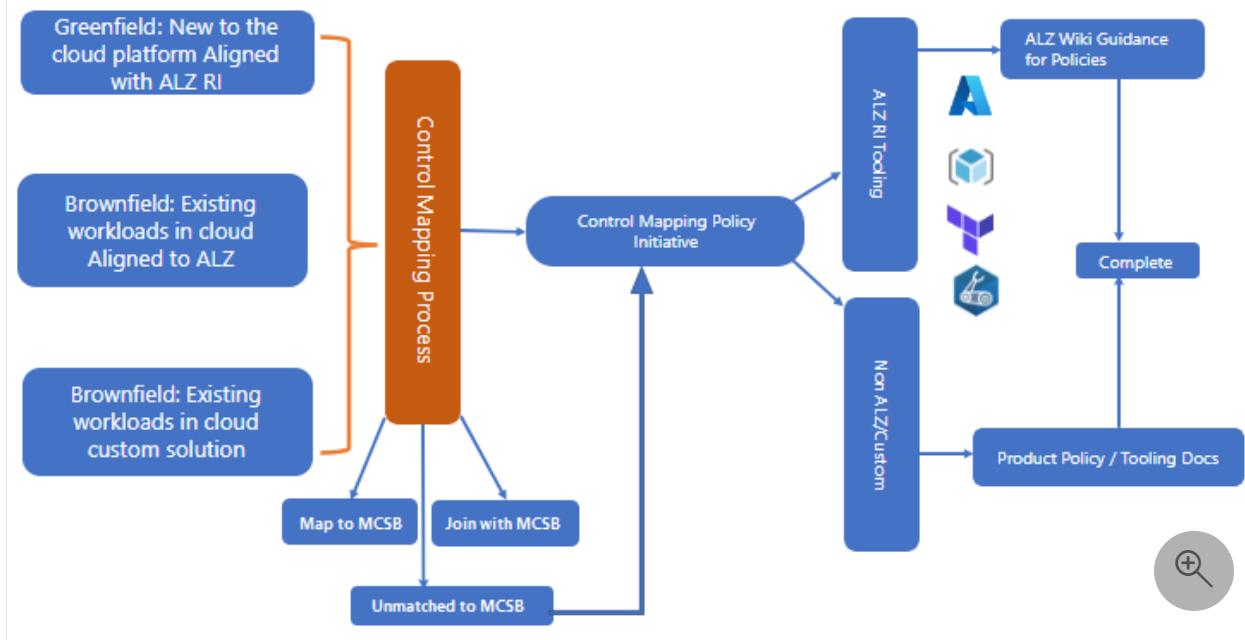
Mapping the required controls to Microsoft cloud security benchmark can greatly expedite secure Azure onboarding experience. Microsoft cloud security benchmark provides a canonical set of cloud-centric technical security controls based on widely used compliance control frameworks such as NIST, CIS, PCI. There are built-in regulatory compliance initiatives already available. If you're interested in a specific compliance domain, refer to [Regulatory compliance built-in initiatives](#).

ⓘ Note

The control mappings between Microsoft cloud security benchmark and industry benchmarks, such as CIS, NIST, and PCI, only indicate that a specific Azure feature can be used to fully or partially address a control requirement defined in these industry benchmarks. You should be aware that such implementation does not necessarily translate to the full compliance of the corresponding controls in these industry benchmarks.

The following diagram shows the process flow of control mapping:

Control mapping process with Azure landing zone integration



Control mapping steps

1. Identify the controls required.
2. Map required controls to Microsoft cloud security benchmark.
3. Identify the controls not mapped with Microsoft cloud security benchmark and respective policies.
4. Perform platform and service level assessment.
5. Implement guardrails with policy initiatives using Azure landing zone tooling, native tooling, or third-party tooling.

💡 Tip

You should review the guidance for how to **tailor the Azure landing zone architecture** to support your control mapping requirements.

1. Identify the controls required

Gather all existing and required lists of compliance controls from the Security team. If the list doesn't exist, capture the control requirements in an Excel spreadsheet. Please use the format below as guidance to build the list. A list would consist of controls from one or many compliance frameworks. Use the [Security control mapping template](#) to capture required controls and related frameworks.

Control Objectives				Microsoft Cloud Security Benchmark (MCSB) Controls			
Control Unique ID No	Topic	CSA CCM v3 Control	Company's Cloud Security Requirement	MCSB Mapping	MCSB Guidance	MCSB Policy	Relevant Azure Security Baseline
CID-AP1	Application Security	AIS-01	Contoso must deploy Web Application Firewalls according to current corporate guidelines • Harden applications against industry baselines • Implement a standardized application patching process	NS-4 PV-7	<p>Protect Azure resources against attacks from external networks, including distributed denial of service (DDoS) Attacks, application specific attacks, and unsolicited and potentially malicious internet traffic. Azure includes native capabilities for this:</p> <ul style="list-style-type: none"> - Use Azure Firewall to protect applications and services against potentially malicious traffic from the Internet and other external locations. - Use Web Application Firewall (WAF) capabilities in Azure Application Gateway, Azure Front Door, and Azure Content Delivery Network (CDN) to protect your applications, services, and APIs against application layer attacks. - Protect your assets against DDoS attacks by enabling DDoS standard protection on your Azure virtual networks. - Use Azure Security Center to detect misconfiguration risks related to the above. <p>Rapidly deploy software updates to remediate software vulnerabilities in operating systems and applications.</p> <p>Use a common risk scoring program (for example, Common Vulnerability Scoring System) or the default risk ratings provided by your third-party scanning tool and tailor to your environment, taking into account which applications present a high security risk and which ones require high uptime.</p> <p>Use Azure Automation Update Management or a third-party solution to ensure that the most recent security</p>	<p>Adaptive network hardening recommendations should be applied on Internet facing virtual machines</p> <p>All internet traffic should be routed via your deployed Azure Firewall</p> <p>All network ports should be monitored on network security groups</p> <p>IP Forwarding should be disabled on virtual machines</p> <p>Authorized IP ranges should be defined on Kubernetes Services</p> <p>Azure Cosmos DB accounts should have firewall rules</p> <p>Azure DDoS Protection Standard should be enabled</p> <p>Azure Key Vault should disable public network access</p> <p>Internet-facing virtual machines should be protected with network security groups</p> <p>IP Forwarding on your virtual machine should be disabled</p> <p>Management ports of virtual machines should be protected with just-in-time network access control</p> <p>Storage accounts should restrict network access</p> <p>Subnets should be associated with a Network Security Group</p> <p>Web Application Firewall (WAF) should be enabled for Application Gateway</p> <p>Web Application Firewall (WAF) should be enabled for Azure Front Door Service service</p> <p>Ensure that 'Java version' is the latest, if used as a part of the API app</p> <p>Ensure that 'Java version' is the latest, if used as a part of the Function app</p> <p>Ensure that 'Java version' is the latest, if used as a part of the Web app</p> <p>Ensure that 'PHP version' is the latest, if used as a part of the API app</p> <p>Ensure that 'PHP version' is the latest, if used as a part of the WEB app</p> <p>Ensure that 'Python version' is the latest, if used as a part of the API app</p> <p>Ensure that 'Python version' is the latest, if used as a part of the Function app</p> <p>Ensure that 'Python version' is the latest, if used as a part of the Web app</p> <p>System updates on virtual machine scale sets should be installed</p> <p>System updates should be installed on your machines</p>	Azure security baseline for App Service Microsoft Docs SOC 2 Audit Report 

A sample of formalized controls list.

2. Map the controls to Microsoft cloud security benchmark and create set of custom controls

For each control you've captured, use appropriate control titles, domain categories, and guidance/description to identify related controls. Align the intent of each control as close as possible and note the deviance or gaps in the spreadsheet.

You can also use common frameworks that are mapped to both your organization's and Microsoft cloud security benchmark where they exist. For example, if both yours and Microsoft cloud security benchmark controls are already mapped to NIST 800-53 r4 or CIS 7.1, you could join the data sets together on that pivot. Intermediate common frameworks can be found in the [resources section](#)

Contoso Control Objectives			
Unique ID No	Topic	CSA CCM v3 Control	Cloud Security Requirement
CID-AP1	Application Security	AIS-01	Contoso must deploy Web Application Firewalls according to current corporate guidelines • Harden applications against industry baselines • Implement a standardized application patching process

Single control mapping example: Your organization's control objectives

The table above shows one of the unique control objectives with key words highlighted.

In this example, we can look at the existing categorization of a given control 'Application Security' to identify it as an application-related control. The content in the requirement field is to implement **application firewalls** and to **harden and patch their applications**. Looking at the Microsoft cloud security benchmark controls and guidance for a proper match, we can see that there are many controls that might apply and map appropriately.

To quickly search a given version of the Microsoft cloud security benchmark, we provide [Excel download files](#) for each release that can be quickly searched by control ID or part of the description verbiage. In this step, the process identifies and maps controls that are covered under Microsoft cloud security benchmark.

3. Identify the controls not mapped with Microsoft cloud security benchmark and respective policies

Any identified controls that might not map directly should be marked as needing mitigating automation, and a custom policy or automation script should be developed in the guardrail implementation process.

Tip

AzAdvertiser [↗](#) is a community driven tool endorsed by the Cloud Adoption Framework. It can help you discover policies that are built-in, from Azure landing zones or from the [community Azure Policy repo](#) [↗](#) in a single place.

4. Perform platform and service level assessment

Once you have your controls and objectives clearly mapped to Microsoft cloud security benchmark and have gathered the supporting information on responsibility, guidance, and monitoring, the IT security office or supporting organization must review all provided information in an official platform assessment.

This platform assessment will determine if the Microsoft cloud security benchmark meets the minimum threshold for usage and if it can meet all security and compliance requirements imposed by the regulations.

If there are gaps identified, you can still use Microsoft cloud security benchmark but might need to develop mitigating controls until these gaps are closed and the benchmark can release updates to address them. In addition, you can map the custom controls by creating a [policy definition](#) and optionally adding to an [initiative definition](#).

Checklists for approval

1. Security team has approved the Azure platform for usage.
2. You'll need to join an individual Microsoft cloud security benchmark service baseline Excel to the previously completed, platform-level control mappings.

- Add in columns to accommodate the assessment like: coverage, enforcement, effects allowed.

3. Perform a line-by-line analysis of the resulting service baseline assessment template:

- For each control objective, indicate:
 - If it can be met by the service or a risk.
 - Risk value, if any.
 - Status of review for that line item.
 - Needed mitigating controls, if any.
 - What Azure Policy can enforce/monitor the control.
- Where there are gaps in monitoring or enforcement for the service and control:
 - Report to the Microsoft cloud security benchmark team to close gaps in content, monitoring, or enforcement.
- For any areas that don't meet your requirements, note the risk involved if you choose to exempt that requirement, the impact, and if it's acceptable to approve or if you're blocked due to the gap.

4. Service status is determined:

- Either the service meets all requirements, or that the risk is acceptable and is placed on an allowlist to be used after guardrails are in place.
- OR, the service gaps are too large / risk is too large and service is placed on a blocklist. It can't be used until gaps are closed by Microsoft.

Inputs - platform level

- Service assessment template (Excel)
- Control objectives to Microsoft cloud security benchmark mapping
- Target service

Outputs - platform level

- Completed service assessment (Excel)
- Mitigating controls
- Gaps
- Approval/non-approval for service usage

After the approval from your internal security/audit team that the platform and core services meet their needs, you need to implement the agreed upon appropriate monitoring and guardrails. During the mapping and assessment process, if there were mitigating controls that extend beyond Microsoft cloud security benchmark, then built-in controls or Azure Policy will need to be implemented using [policy definitions](#) and optionally adding to an [initiative definition](#).

Checklist - service level

1. Summarize the policies that were identified as required as an output of the platform assessment and service assessments.
2. Develop any needed custom policy definitions to support mitigating controls/gaps.
3. Create custom policy initiative.
4. Assign the policy initiative with Azure landing zone tooling, native tooling, or third-party tooling.

Inputs - service level

- Completed service assessment (Excel)

Outputs - service level

- Custom policy initiative

5. Implement guardrails using Azure landing zone or native tools

The following sections describe the process of identifying, mapping, and implementing regulatory compliance-related controls as part of Azure landing zone deployment. Deployment covers policies that are aligned with Microsoft cloud security benchmark for platform-level security controls.

💡 Tip

As part of the Azure landing zone accelerators ([Portal](#), [Bicep](#) & [Terraform](#)), we assign the Microsoft cloud security benchmark policy initiative to Intermediate Root Management Group by default.

You can learn about [policies assigned as part of an Azure landing zone Accelerator deployment](#).

Implementation policy guidance

Depending on your control objectives, you might need to create custom [policy definitions](#), [policy initiative definitions](#) and [policy assignments](#).

Refer to the following guidance for each accelerator implementation option.

Azure landing zone accelerator portal

When using the [Azure landing zone portal-based experience](#):

- [Create custom security policies in Microsoft Defender for Cloud](#)
- [Tutorial: Create a custom policy definition](#)
- [Assign Azure Policy or policy initiatives](#)

Azure Resource Manager with AzOps

When using the [Resource Manager templates](#) with the [AzOps Accelerator](#), refer to the deployment article to learn how to operate the Azure platform using infrastructure as code.

- [Adding Custom Azure Policy definitions and initiatives](#)
- [Assigning Azure Policy](#)

Terraform module

When using the [Azure landing zones Terraform module](#), refer to the repository wiki for guidance on how to manage additional policy definitions and assignments.

- [Adding Custom Azure Policy definitions and assignments](#)
- [Assigning a built-in Azure Policy](#)
- [Expand built-in archetype definitions](#)

Bicep

When using the Azure landing zones Bicep implementation, learn how to [create your own policy definitions and assignments](#).

- [Adding Custom Azure Policy definitions and initiatives](#)
- [Assigning Azure Policies](#)

Implement custom policies when not using an Azure landing zones implementation

Azure portal

When using the Azure portal, refer to the following articles.

- [Create custom security policies in Microsoft Defender for Cloud](#)
- [Create a custom policy definition](#)
- [Create and manage policies to enforce compliance](#)
- [Assign policy initiatives](#)

Azure Resource Manager templates

When using the Resource Manager templates, refer to the following articles.

- [Create a custom policy definition](#)
- [Assign policy initiatives](#)
- [Create a policy assignment to identify non-compliant resources by using an ARM template](#)
- [Bicep and Resource Manager policy definition template reference](#)
- [Bicep and Resource Manager set \(initiative\) template reference](#)
- [Bicep and Resource Manager policy assignment template reference](#)

Terraform

When using Terraform, refer to the following articles.

- [Adding custom Azure policy definitions and initiatives ↗](#)
- [Adding Azure Policy set Definition ↗](#)
- [Assigning management group policy ↗](#)
- [Assigning Azure Policy or policy initiative ↗](#)

Bicep

When using the Bicep templates, refer to the following articles.

- [Quickstart: Create a policy assignment to identify non-compliant resources by using a Bicep file](#)
- [Bicep and Resource Manager policy definition template reference](#)
- [Bicep and Resource Manager policy set \(initiative\) template reference](#)

- Bicep and Resource Manager policy assignment template reference

Guidance for using Microsoft Defender for Cloud

[Microsoft Defender for Cloud](#) continually compares the configuration of your resources with requirements in industry standards, regulations, and benchmarks. The regulatory compliance dashboard provides insight into your compliance posture. Learn more about [improving your regulatory compliance](#).

Frequently asked questions

We're using a framework not mapped to Microsoft cloud security benchmark, how can I still onboard our control objectives?

We provide Microsoft cloud security benchmark mappings to many of the most in-demand industry frameworks. However, for the controls that are currently not covered, a manual mapping exercise is needed. In these instances, refer to our steps for performing a manual control mapping.

[Example] We need to meet Canada Federal Protected B (PBMM) compliance, and Microsoft cloud security benchmark doesn't yet have a mapping to PBMM. To bridge this mapping, you can find a shared framework mapping such as NIST SP 800-53 R4 that's available and mapped to both PBMM and MCSB v2. Using this common framework, you can understand what recommendations and guidance you must follow in Azure to meet your desired framework.

Our control objectives aren't covered by the Microsoft cloud security benchmark controls, how can I unblock them from onboarding?

Microsoft cloud security benchmark is focused on Azure technical controls. Objective areas surrounding non-technical items such as training, or for items that aren't direct technical security, such as data center security, are omitted by design. These items can be marked as Microsoft responsibility, and evidence of compliance can be provided from Microsoft cloud security benchmark content or Microsoft audit reports. If you find that the objective truly is a technical control, then create a mitigating control in-addition

to the base for tracking, and send a request to MCSBteam@microsoft.com to address the missing controls in future releases.

Resources

[Service Trust Portal ↗](#)

[Cloud Security Alliance](#)

[Datacenter Security Overview](#)

[Financial Services Overview ↗](#)

[Financial Institution Risk Assessment Overview ↗](#)

[Service Level Agreement ↗](#)

Incorporate Zero Trust practices in your landing zone

Article • 12/20/2023

[Zero Trust](#) is a security strategy in which you incorporate products and services into your design and implementation to adhere to the following security principles:

- **Verify explicitly:** always authenticate and authorize access based on all available data points.
- **Use least-privilege access:** limit users to just-enough access, and use tools to provide just-in-time access with considerations to adaptive risk-based policies.
- **Assume breach:** minimize the blast radius and segment access, proactively look for threats, and continually improve defenses.

If your organization adheres to the Zero Trust strategy, you should incorporate Zero Trust-specific deployment objectives into your landing zone design areas. Your landing zone is the foundation of your workloads in Azure, so it's important to prepare your landing zone for Zero Trust adoption.

This article provides guidance for integrating Zero Trust practices into your landing zone and explains where adherence to Zero Trust principles requires solutions outside your landing zone.

Zero Trust pillars and landing zone design areas

When you implement Zero Trust practices in your Azure landing zone deployment, you should begin by considering the Zero Trust guidance for each landing zone design area.

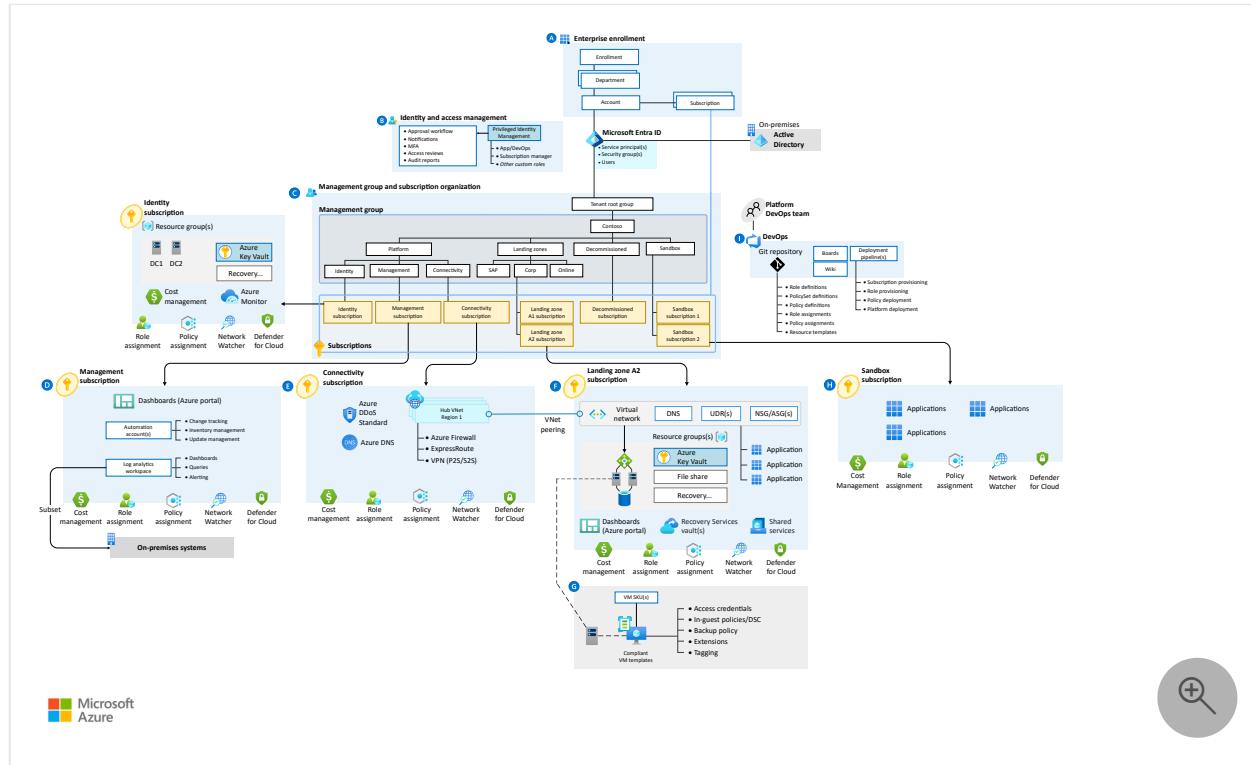
For considerations about designing a landing zone and guidance for critical decisions in each area, see [Azure landing zone design areas](#).

The Zero Trust model has pillars that are organized by concepts and deployment objectives. For more information, see [Deploying Zero Trust solutions](#).

These pillars provide specific deployment objectives that help organizations align with Zero Trust principles. These objectives go beyond technical configurations. For example, the networking pillar has a deployment objective for network segmentation. The objective doesn't provide information on how to configure isolated networks in Azure

but instead offers guidance for creating the architecture pattern. There are other design decisions to consider when you implement a deployment objective.

The following diagram shows the landing zone design areas.



The following table correlates the Zero Trust pillars to the design areas shown in the architecture.

[Expand table](#)

Legend	Landing zone design area	Zero Trust pillar
A	Azure billing and Microsoft Entra tenant	Identity pillar
B	Identity and access management	Identity pillar, Applications pillar, Data pillar
C	Resource organization	Identity pillar
D E	Governance	Visibility, automation, and orchestration pillar
D G H	Management	Endpoints pillar, Applications pillar, Data pillar, Infrastructure pillar

Legend	Landing zone design area	Zero Trust pillar
E	Network topology and connectivity	Networks pillar
F	Security	All Zero Trust pillars
I	Platform automation and DevOps	Visibility, automation, and orchestration pillar

Not all of the Zero Trust deployment objectives are part of a landing zone. Many Zero Trust deployment objectives are for designing and releasing individual workloads to Azure.

The following sections review each pillar and provide considerations and recommendations for implementing deployment objectives.

Secure identity

For information about deployment objectives for securing identity, see [Securing identity with Zero Trust](#). To implement these deployment objectives, you can apply identity federation, conditional access, identity governance, and real-time data operations.

Identity considerations

- You can use [Azure landing zone reference implementations](#) to deploy resources that extend your existing identity platform into Azure, and manage the identity platform by implementing Azure best practices.
- You can configure many of the controls for Zero Trust practices in your Microsoft Entra tenant. You can also control access to Microsoft 365 and other cloud services that use Microsoft Entra ID.
- You must plan configuration requirements beyond what's in your Azure landing zone.

Identity recommendations

- Develop a plan for managing identities in Microsoft Entra ID that go beyond Azure resources. For example, you can use:
 - Federation with on-premises identity systems.
 - Conditional access policies.
 - User, device, location, or behavior information for authorization.

- Deploy your Azure landing zone with separate subscriptions for identity resources, like domain controllers, so you can better secure access to resources.
- Use Microsoft Entra managed identities where possible.

Secure endpoints

For information about deployment objectives for securing endpoints, see [Secure endpoints with Zero Trust](#). To implement these deployment objectives, you can:

- Register endpoints with cloud identity providers to provide access to resources solely through cloud-managed compliant endpoints and apps.
- Enforce data loss prevention (DLP) and access control for both corporate devices and personal devices that are enrolled in *bring your own device* (BYOD) programs.
- Monitor the device risk for authentication with endpoint threat detection.

Endpoint considerations

- Endpoint deployment objectives are for end-user compute devices, such as laptops, desktop computers, and mobile devices.
- As you adopt Zero Trust practices for endpoints, you must implement solutions in Azure and outside Azure.
- You can use tools, such as Microsoft Intune and other device management solutions, to realize deployment objectives.
- If you have endpoints in Azure, such as in Azure Virtual Desktop, you can enroll the client experience in Intune, and apply Azure policies and controls to restrict access to the infrastructure.

Endpoint recommendations

- Develop a plan for managing endpoints with Zero Trust practices, in addition to your plans to implement an Azure landing zone.
- For other information about devices and servers, see [Secure infrastructure](#).

Secure applications

For information about deployment objectives for securing applications, see [Secure applications with Zero Trust](#). To implement these deployment objectives, you can:

- Use APIs to gain visibility into applications.
- Apply policies to protect sensitive information.
- Apply adaptive access controls.
- Limit the reach of shadow IT.

Application considerations

- The deployment objectives for applications focus on managing both third-party and first-party applications in your organization.
- The objectives don't address securing application infrastructure. Instead, they address securing the consumption of applications, especially cloud applications.
- The Azure landing zone practices don't provide detailed controls for application objectives. These controls are configured as part of the application configuration.

Application recommendations

- Use [Microsoft Defender for Cloud Apps](#) to manage access to applications.
- Use the standardized policies included in Defender for Cloud Apps to enforce your practices.
- Develop a plan to onboard your applications to your practices for application access. Don't trust applications that your organization hosts any more than you trust third-party applications.

Secure data

For information about deployment objectives for securing data, see [Secure data with Zero Trust](#). To implement these objectives, you can:

- Classify and label data.
- Enable access control.
- Implement data loss protection.

For information about logging and managing data resources, see [Azure landing zone reference implementations](#).

A Zero Trust approach involves extensive controls for data. From an implementation stand point, [Microsoft Purview](#) provides tools for data governance, protection, and risk management. You can use Microsoft Purview as part of a [cloud-scale analytics](#) deployment to provide a solution that you can implement at scale.

Data considerations

- In accordance with the landing zone subscription democratization principle, you can create access and network isolation for data resources, and also establish logging practices.

There are [policies](#) in the reference implementations for logging and managing data resources.

- You need other controls beyond securing Azure resources to meet the deployment objectives. Zero Trust data security involves classifying data, labeling it for sensitivity, and controlling data access. It also extends beyond database and file systems. You need to consider how to protect data in Microsoft Teams, Microsoft 365 Groups, and SharePoint.

Data recommendations

- [Microsoft Purview](#) provides tools for data governance, protection, and risk management.
- Implement Microsoft Purview as part of a [cloud-scale analytics](#) deployment to implement your workload at scale.

Secure infrastructure

For information about deployment objectives for securing infrastructure, see [Secure infrastructure with Zero Trust](#). To implement these objectives, you can:

- Monitor abnormal behavior in workloads.
- Manage infrastructure identities.
- Limit human access.
- Segment resources.

Infrastructure considerations

- Infrastructure deployment objectives include:

- Managing Azure resources.
- Managing operating system environments.
- Accessing systems.
- Applying workload-specific controls.
- You can use the landing zone subscription model to create clear security boundaries to Azure resources, and assign limited permissions as needed at the resource level.
- Organizations need to organize their workloads for management.

Infrastructure recommendations

- Use the standard [Azure landing zone policies](#) to block noncompliant deployments and resources, and to enforce logging patterns.
- Configure [Privileged Identity Management](#) in Microsoft Entra ID to provide just-in-time access to highly privileged roles.
- Configure [just-in-time access](#) in Defender for Cloud for your landing zone to restrict access to virtual machines.
- Create a plan to monitor and manage individual workloads that are deployed in Azure.

Secure networks

For information about deployment objectives for securing networks, see [Secure networks with Zero Trust](#). To implement these objectives, you can:

- Implement network segmentation.
- Use cloud-native filtering.
- Implement least-access privilege.

Network considerations

- To ensure that your platform resources support the Zero Trust security model, you must deploy firewalls that are capable of HTTPS traffic inspection and isolate identity and management network resources from the central hub.
- In addition to the networking resources in the connectivity subscription, you need to create plans to micro-segment individual workloads in their spoke virtual

networks. For example, you can define traffic patterns and create fine-grained network security groups for each workload network.

Network recommendations

- Use the following Zero Trust-specific deployment guides to deploy your Azure landing zone:
 - [Azure landing zone portal accelerator deployment with Zero Trust network principles ↗](#)
 - [Deploy networking with Zero Trust network principles ↗](#)
 - [Azure landing zone Terraform deployment with Zero Trust network principles ↗](#)
- For information about how to apply Zero Trust principles to the application delivery, see [Zero Trust network for web applications](#).
- For information about how to create a plan for workload networking, see [Zero Trust deployment plans with Azure](#).

Visibility, automation, and orchestration

For information about deployment objectives for visibility, automation, and orchestration, see [Visibility, automation, and orchestration with Zero Trust](#). To implement these objectives, you can:

- Establish visibility.
- Enable automation.
- Enable additional controls by practicing continual improvement.

Visibility, automation, and orchestration considerations

- The [Azure landing zone reference implementations](#) contain deployments of [Microsoft Sentinel](#) that you can use to quickly establish visibility in your Azure environment.
- The reference implementations provide policies for Azure logging, but additional integration is needed for other services.
- You should configure automation tools, like Azure DevOps and GitHub, to send signals.

Visibility, automation, and orchestration recommendations

- Deploy Microsoft Sentinel as part of your Azure landing zone.
- Create a plan to integrate signals from Microsoft Entra ID and tools into Microsoft 365 to your Microsoft Sentinel workspace.
- Create a plan for conducting threat-hunting exercises and continual security improvements.

Next steps

- [Security in the Microsoft Cloud Adoption Framework for Azure](#)
 - [Apply Zero Trust principles to Azure services](#)
 - [Evaluate your Zero Trust security posture ↗](#)
-

Feedback

Was this page helpful?



Design area: Management for Azure environments

Article • 03/21/2023

This design area establishes a foundation for operations management across your Azure, hybrid, or multicloud environments. You can enhance your learning later on using the operations guidance outlined in [Manage methodology](#) of the Cloud Adoption Framework.

Design area review

Involved roles or functions: This design area is led by [central IT](#) or [cloud operations](#), specifically the [security architects within that team](#). The [cloud platform](#) and [cloud center of excellence](#) will likely be required to define and implement the technical requirements coming from this exercise. More advanced operations guardrails might also require support from [cloud governance](#).

Scope: The goal of this exercise is to understand operations management requirements and implement those requirements consistently across all workloads in your cloud platform. The primary scope of this exercise focuses on operations tooling. You'll use operations tooling to manage the collective portfolio of workloads with a set of common tools and processes. This initial set of operations tooling is also referred to as your operations baseline.

Out of scope: You can use the operations baseline defined in this exercise consistently across all workloads. The operations baseline can also be expanded with other tools and processes as outlined in the [Manage methodology](#) of the Cloud Adoption Framework. Doing so helps to improve operations for specific technology platforms or individual workloads.

You can also use the operations baseline with the Azure Well-Architected Framework and Microsoft Azure Well-Architected Review to improve the operations and architecture of individual workloads you deploy within your cloud environment. However, any advanced operations, tech platform operations, or workload operations are out of scope for this exercise.

Design area overview

For stable, ongoing operations in the cloud, a management baseline is required to provide visibility, operations compliance, and protect and recover capabilities.

The management design area focuses on the considerations and recommendations for landing zone design decisions. Also, the [Manage methodology](#) of the Cloud Adoption Framework provides further in-depth guidance for holistic management processes and tools.

Operations baseline

Use the following operations items to evaluate which operations management tooling you need to include in your operations baseline.

Scope	Context
Inventory & visibility	<p>As cloud environments are implemented and scaled out, management controls that span the environment become increasingly important.</p> <p>No matter the services that are running on top of the landing zone, the management of fundamental elements of the platform is necessary to ensure stable, ongoing operations.</p> <p>These management tools should scale as the environments do.</p> <p>They can include a mix of first-party and third-party tools, depending on your existing investments.</p>
Operational Compliance	<p>Requirements for patching and managing configuration drift.</p> <p>Requirements for automatic or centralize resource optimization and sizing.</p> <p>Requirements for workloads that should only be optimized or resized by the assigned workload teams.</p> <p>Processes for ensuring completion of their regular optimization efforts.</p>
Protect & Recover	<p>Your organization needs to design suitable, platform-level capabilities that application workloads can depend on for a basic level of business continuity and disaster recovery.</p> <p>Specifically, these application workloads have requirements related to recover time objective (RTO) and recovery point objective (RPO). Be sure that you capture disaster recovery (DR) requirements to identify and address needs for advanced operations.</p>

Advanced operations

Use the following advanced operations items as discussion points within your cloud architecture and operations teams. These discussions let you explore and agree on the requirements and features to include in your management design.

Scope	Context
-------	---------

Scope	Context
Platform management	<p>When evaluating supported workloads, it's common for those workloads to have dependencies on shared platforms, like SAP, WVD, AVS, SQL, and so on. When technology platforms are used by multiple workloads, advanced operations can't be delegated to a single workload team. In these instances, centralized operations teams need a plan for the ongoing operations of those shared technology platforms. These responsibilities require extra tooling beyond the operations baseline that supports the overall cloud environment.</p>
Workload management	<p>Workloads built on top of the landing zone platform might have specific management requirements in addition to the tools and processes put in place for the platform services.</p> <p>These requirements should be considered in the context of the platform management to ensure that additions or exceptions are known and documented. It's also important to look at these requirements in the broader context. Often, what is thought to be a requirement for a single workload can become a common pattern. Consider these situations as part of the overall platform toolset to avoid unnecessary duplication of effort.</p> <p>For further information on considerations for workload-specific management, review the operational excellence of the Azure Well-Architected Framework.</p>

Inventory and visibility considerations

Article • 05/07/2024

As your organization designs and implements your cloud environment, the basis for your platform management and platform services monitoring is a key consideration. To ensure a successful cloud adoption, you must structure these services to meet the needs of your organization as your environment scales.

The cloud operating model decisions you make in early planning phases directly influence how management operations are delivered as part of your landing zones. The degree to which management is centralized for your platform is a key example.

Use the guidance in this article to consider how you should approach inventory and visibility in your cloud environment.

Basic inventory considerations

- Consider using tools such as an Azure Monitor Log Analytics workspace as administrative boundaries.
- Determine which teams should use the system-generated logs from the platform and who needs access to those logs.

Consider the following items related to logging data to inform what types of data you might want to collate and use.

[] Expand table

Scope	Context
Application-centric platform monitoring	<p>Include both hot and cold telemetry paths for metrics and logs, respectively.</p> <p>Operating system metrics, such as performance counters and custom metrics.</p> <p>Operating system logs, such as:</p> <ul style="list-style-type: none">• Internet Information Services• Event Tracing for Windows, and syslogs• Resource health events
Security audit logging	<p>Aim to achieve a horizontal security lens across your organization's entire Azure estate.</p> <ul style="list-style-type: none">• Potential integration with on-premises security information and event management (SIEM) systems such as ArcSight or

Scope	Context
	<p>the Onapsis security platform</p> <ul style="list-style-type: none"> • Potential integration with software as a service (SaaS) offerings like ServiceNow • Azure activity logs • Microsoft Entra audit reports • Azure diagnostic services, logs, and metrics, Azure Key Vault audit events, network security group (NSG) flow logs, and event logs • Azure Monitor, Azure Network Watcher, Microsoft Defender for Cloud, and Microsoft Sentinel
Azure data retention thresholds and archiving requirements	<ul style="list-style-type: none"> • The default retention period for Azure Monitor Logs is 30 days, with a maximum analytics retention of two years and archive of seven years. • The default retention period for Microsoft Entra reports (premium) is 30 days. • The default retention period for the Azure Activity logs and Application Insights logs is 90 days.
Operational requirements	<ul style="list-style-type: none"> • Operational dashboards with native tools such as Azure Monitor Logs or third-party tooling • Use of centralized roles to control privileged activities • Managed identities for Azure resources](/Azure/active-directory/managed-identities-Azure-resources/overview) for access to Azure services • Resource locks to protect from editing and deleting resources

Visibility considerations

- Which teams need to receive alert notifications?
- Do you have groups of services that need multiple teams to be notified?
- Do you have existing Service Management tools in place that you need to send alerts to?
- Which services are considered business critical and require high priority notifications of issues?

Inventory and visibility recommendations

- Use a single [monitor logs workspace](#) to manage platforms centrally, except where Azure role-based access control (Azure RBAC), data sovereignty requirements, and data retention policies mandate separate workspaces. Centralized logging is critical

to the visibility required by operations management teams and drives reports about change management, service health, configuration, and most other aspects of IT operations. Focusing on a centralized workspace model reduces administrative effort and the chances for gaps in observability.

- Export logs to Azure Storage if your log retention requirements exceed seven years. Use immutable storage with a write-once, read-many policy to make data non-erasable and non-modifiable for a user-specified interval.
- Use Azure Policy for access control and compliance reporting. Azure Policy lets you enforce organization-wide settings to ensure consistent policy adherence and fast violation detection. For more information, see [Understand Azure Policy effects](#).
- Use Network Watcher to proactively monitor traffic flows through [Network Watcher NSG flow logs v2](#). [Traffic Analytics](#) analyzes NSG flow logs to gather deep insights about IP traffic within virtual networks. It also provides critical information you need for effective management and monitoring, such as:
 - Most communicating hosts and application protocols
 - Most conversing host pairs
 - Allowed or blocked traffic
 - Inbound and outbound traffic
 - Open internet ports
 - Most blocking rules
 - Traffic distribution per an Azure datacenter
 - Virtual network
 - Subnets
 - Rogue networks
- Use [resource locks](#) to prevent accidental deletion of critical shared services.
- Use [deny policies](#) to supplement Azure role assignments. Deny policies help prevent resource deployments and configurations that don't meet defined standards by blocking requests from being sent to resource providers. Combining deny policies and Azure role assignments ensures that you have appropriate guardrails in place to control *who* can deploy and configure resources and *which* resources they can deploy and configure.
- Include [service](#) and [resource](#) health events as part of your overall platform monitoring solution. Tracking service and resource health from the platform perspective is an important component of resource management in Azure.
- Don't send raw log entries back to on-premises monitoring systems. Instead, adopt the principle that *data born in Azure stays in Azure*. If you require on-

premises SIEM integration, send [critical alerts](#) instead of logs.

Azure landing zone accelerator and management

The Azure landing zone accelerator includes opinionated configuration to deploy key Azure management capabilities that help your organization quickly scale and mature.

The Azure landing zone accelerator deployment includes key management and monitoring tools like:

- A Log Analytics workspace and Automation account
- Microsoft Defender for Cloud monitoring
- Diagnostic settings for activity logs, virtual machines, and platform as a service (PaaS) resources sent to Log Analytics

Centralized logging in the Azure landing zone accelerator

In the context of the Azure landing zone accelerator, centralized logging is primarily concerned with platform operations.

This emphasis doesn't prevent use of the same workspace for VM-based application logging. Within a workspace configured in resource-centric access control mode, granular Azure RBAC is enforced, which ensures that your application teams only have access to the logs from their resources.

In this model, application teams benefit from the use of existing platform infrastructure as it reduces their management overhead.

For non-compute resources, like web apps or Azure Cosmos DB databases, your application teams can use their own Log Analytics workspaces. They can then route diagnostics and metrics to those workspaces.

Next step

[Monitor your Azure platform landing zone components](#)

Feedback

Was this page helpful?

 Yes

 No

Monitor Azure platform landing zone components

Article • 04/19/2024

Monitor your Azure platform landing zone components to ensure availability, reliability, security, and scalability. Monitoring your components enables your organization to:

- Promptly detect and resolve issues, optimize resource utilization, and proactively address security threats.
- Continuously monitor performance and health, which helps your organization minimize downtime, optimize costs, and ensure efficient operation.
- Facilitate capacity planning, which allows your organization to anticipate resource needs and scale the platform accordingly.

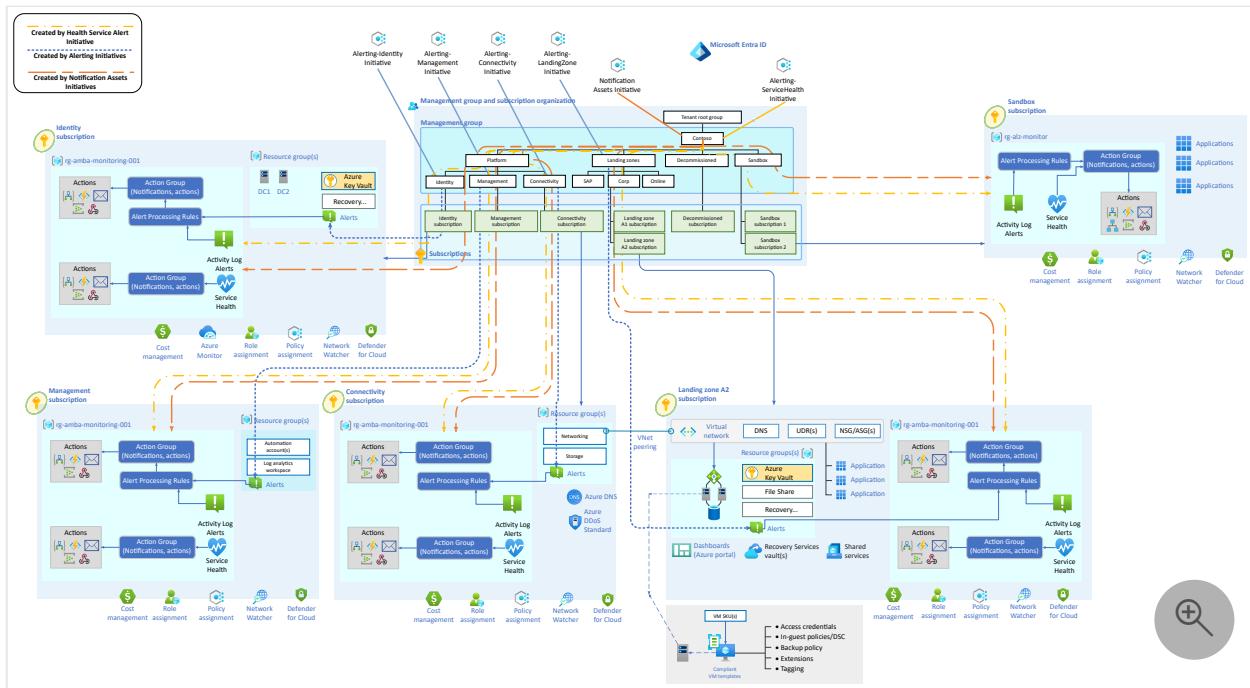
Monitor the platform landing zone services to maintain a stable and secure environment, maximize performance, and effectively meet the evolving needs of your business.

The following video discusses the solution that's described in this article and demonstrates how to deploy Azure Monitor.

<https://www.youtube-nocookie.com/embed/xeuxetAfHRg?si=1dmmOylAkdf8nMgV>

Azure landing zone monitoring guidance

Baseline metric, activity log, and log query alerts are available for landing zone platform components and other selected landing zone components. These alerts ensure consistent alerting and monitoring for your Azure landing zone. They're based on Microsoft-recommended practices for proactive monitoring, such as setting up alerts, thresholds, and notifications for timely problem detection and response. Use the following guidance to achieve real-time visibility into the performance, utilization, and security of your platform landing zone implementation. Proactively address issues, optimize resource allocation, and ensure a reliable and secure environment.



[Download a Visio file](#) of this architecture.

The following subsets of Azure components have one or more alerts defined:

- Azure ExpressRoute
- Azure Firewall
- Azure Virtual Network
- Azure Virtual WAN
- Azure Monitor Log Analytics workspace
- Azure Private DNS zone
- Azure Key Vault
- Azure Virtual Machine
- Azure Storage account

For more information, see [Alert details](#).

To ensure that your organization's resources are properly monitored and secured, you also need to properly configure alerts and implement appropriate processes to respond to alerts. Configure action groups with the appropriate notification channels and test the alerts to ensure that they work as expected. In accordance with the Cloud Adoption Framework principle of subscription democratization, configure at least one action group for each subscription so that relevant staff is notified of alerts. As a minimum form of notification, the action group should include an email notification channel. If you use Azure Monitor alert processing rules to route alerts to one or more action groups, note that service health alerts don't support alert processing rules. Configure service health alerts directly with the action group.

In accordance with [Azure landing zone principles for policy-driven governance](#), a framework solution is available that provides an easy way to scale alerting by using Azure Policy. These policies use the *DeployIfNotExists* effect to deploy relevant alert rules, alert processing rules, and action groups when you create a resource in your Azure landing zone environment, in both platform services and landing zones.

Azure Policy provides a default baseline configuration, but you can configure the policies to suit your needs. If you require alerting on different metrics than what the repository provides, the solution provides a framework for you to develop your own policies for deploying alert rules. For more information, see the [Azure Monitor baseline alerts \(AMBA\) contributor guide](#) and [Introduction to an Azure Monitor deployment](#). The solution is integrated into the Azure landing zone installation experience, so new implementations of Azure landing zone offer you the opportunity to set up baseline alerting at installation time.

Deploy alerts via policies to provide flexibility and scalability, and to ensure that alerts are deployed inside and outside the Azure landing zone scope. Alerts are only deployed when the corresponding resources are created, which avoids unnecessary cost and ensures updated alerts configurations at deployment time. This function coincides with the Azure landing zone principle of policy-driven governance.

Brownfield guidance

In a brownfield scenario, your organization has an existing Azure landing zone implementation, or your organization implemented Azure before an Azure landing zone architecture was available.

This section describes the high-level steps for Azure landing zone baseline monitoring if you have an existing Azure footprint, whether aligned to Azure landing zones or not.

Aligned to an Azure landing zone

Use this process if you have an existing Azure landing zone implementation and you want to use the policy-driven approach.

1. Import relevant policies and initiatives from the AMBA repository.
2. Assign the required policies in your environment.
3. Remediate noncompliant policies.

For more information about policies relevant to your environment and the steps to apply them, see [Determine your management group hierarchy](#).

Not aligned to an Azure landing zone

Use this process if you have a non-Azure landing zone aligned implementation and you want to use the policy-driven approach.

1. Import relevant policies and initiatives from the AMBA repository to the top-most management group from which you wish to assign the policies.
2. Assign the required policies in your environment.
3. Remediate noncompliant policies.

For more information about policies relevant to your environment and the steps to apply them, see [Determine your management group hierarchy](#).

Test the framework

Test policies and alerting before a deployment to production so you can:

- Ensure that your organization's resources are properly monitored and secured.
- Identify issues early to ensure proper functionality, reduce risk, and improve performance.
- Detect and fix problems before they become larger. Avoid false positives or negatives and reduce the risk of costly mistakes.
- Optimize configurations for better performance and avoid performance-related issues in production.

Testing helps you ensure that your alerting and policy configurations meet your organization's requirements and comply with regulations and standards. With regulations in place, you can avoid security breaches, compliance violations, and other risks that can have consequences for your organization.

Testing is an essential step in the development and deployment of alerting and policy configurations, and can help you ensure the security, reliability, and performance of your organization's resources.

For more information, see [Testing approach for Azure landing zones](#).

Note

If you implement alerting by using a different approach, like infrastructure as code (IaC), for example Azure Resource Manager, Bicep, or Terraform, or via the portal, the guidance for alerts, severity, and thresholds that's in the repository still applies for determining which alert rules to configure and for notifications.

Next steps

- [Introduction to deploying Azure Monitor ↗](#)
 - [Business continuity and disaster recovery](#)
-

Feedback

Was this page helpful?



Business continuity and disaster recovery

Article • 02/14/2024

Organization and enterprise application workloads have recovery time objective (RTO) and recovery point objective (RPO) requirements. Effective business continuity and disaster recovery (BCDR) design provides platform-level capabilities that meet these requirements. To design BCDR capabilities, capture platform disaster recovery (DR) requirements.

Design considerations

Consider the following factors when designing BCDR for application workloads:

- Application and data availability requirements:
 - RTO and RPO requirements for each workload.
 - Support for active-active and active-passive availability patterns.
- BCDR as a service for platform-as-a-service (PaaS) services:
 - Native DR and high-availability (HA) feature support.
 - Geo-replication and DR capabilities for PaaS services.
- Support for multiregion deployments for failover, with component proximity for performance.
- Application operations with reduced functionality or degraded performance during an outage.
- Workload suitability for Availability Zones or availability sets:
 - Data sharing and dependencies between zones.
 - Availability Zones compared to availability sets impact on update domains.
 - Percentage of workloads that can be under maintenance simultaneously.
 - Availability Zones support for specific virtual machine (VM) stock-keeping units (SKUs). For example, Azure Ultra Disk Storage requires using Availability Zones.
- Consistent backups for applications and data:
 - VM snapshots.
 - Azure Backup Recovery Services vaults.
 - Subscription limits restricting the number of Recovery Services vaults and the size of each vault.

- Network connectivity if a failover occurs:
 - Bandwidth capacity planning for Azure ExpressRoute.
 - Traffic routing during a regional, zonal, or network outage.
- Planned and unplanned failovers:
 - IP address consistency requirements, and the potential need to maintain IP addresses after failover and failback.
 - Maintaining engineering DevOps capabilities.
 - Azure Key Vault DR for application keys, certificates, and secrets.
- Data residency:
 - Understand the in-country/region guidance for data residency that specifies whether data should be kept within country or regional borders. This guidance affects your design for cross-region replication.
 - Azure regions that reside within the same geography as their enabled set can help with cross-region replication to meet data residency requirements such as tax and law enforcement requirements. For more information, see [Azure cross-region replication](#).

Design recommendations

The following design practices support BCDR for application workloads:

- Employ Azure Site Recovery for Azure-to-Azure VM DR scenarios.

Site Recovery uses real-time replication and recovery automation to replicate workloads across regions. Built-in platform capabilities for VM workloads meet low RPO and RTO requirements. You can use Site Recovery to run recovery drills without affecting production workloads. You can also use Azure Policy to enable replication and to audit VM protection.

- Use native PaaS DR capabilities.

Built-in PaaS features simplify both design and deployment automation for replication and failover in workload architectures. Organizations that define service standards can also audit and enforce the service configuration through Azure Policy.

- Use Azure-native backup capabilities.

Azure Backup and PaaS-native backup features remove the need for third-party backup software and infrastructure. As with other native features, you can set,

audit, and enforce backup configurations with Azure Policy to ensure compliance with organization requirements.

- Use multiple regions and peering locations for ExpressRoute connectivity.

A redundant hybrid network architecture can help ensure uninterrupted cross-premises connectivity if an outage affects an Azure region or peering provider location.

- Avoid using overlapping IP address ranges in production and DR networks.

Production and DR networks that have overlapping IP addresses require a failover process that can complicate and delay application failover. When possible, plan for a BCDR network architecture that provides concurrent connectivity to all sites.

Feedback

Was this page helpful?

 Yes

 No

Operational compliance considerations

Article • 07/05/2023

Throughout your cloud adoption journey, your environments will continue to scale and your number of applications and services will continue to grow. It's important that you put capabilities in place to monitor for deviations from your expected configurations.

Your tools should include automation wherever possible. Automation enables them to scale, covering your growing environment footprint and reducing the risk of gaps in observation.

Monitor for configuration drift

Monitoring your environments for configuration drift is an important part of ensuring stable and consistent operations.

[Azure Policy](#) is valuable within cloud management processes. Azure Policy can audit and remediate Azure resources, and can also audit settings inside a machine. Validation is performed by the Azure Automanage Machine Configuration extension and client. The extension, through the client, validates settings such as:

- Operating system configuration
- Application configuration or presence
- Environment settings

Use this technique as part of your organization's management approach within landing zones, where it can assist help ensure resources stay in line with an expected configuration.

In addition, using Infrastructure as Code can help you monitor for configuration drift, as well as help you keep your landing zone up to date. To learn more, see [Keep your Azure landing zone up to date](#) and [Use infrastructure as code to update Azure landing zones](#).

Learn about [Azure Automanage Machine Configuration](#). Consider how you can use it as part of your landing zone management toolkit.

Update management considerations

- Does your organization currently use any update management tools? Can these tools be extended to cover your cloud environment, or will you need new tools?
- Which teams should be responsible for overseeing update management?

- Do you have groups of resources that share similar update schedules?
- Do you have groups of resources that can't be updated at the same time for business continuity reasons?

Operational compliance recommendations

- Use [Update Management in Azure Automation](#) as a long-term patching mechanism for both Windows and Linux VMs. Enforcing Update Management configurations through Azure Policy ensures that all VMs are included in your patch management regimen. It also provides your application teams with the ability to manage patch deployment for their VMs, and provides visibility and enforcement capabilities to your central IT team across all VMs.
- Use Azure Policy to monitor in-machine virtual machine (VM) configuration drift. Enabling [Azure Automanage Machine Configuration](#) audit capabilities through policy helps your application team workloads to consume feature capabilities immediately with little effort.

Next steps

Learn how your workload teams can use a federated model and operationally maintain their workloads.

[Workloads](#)

Workload management and monitoring

Article • 11/12/2024

This guidance uses a federated model to explain how workload teams can operationally maintain and monitor their workloads.

Workload management and monitoring design considerations

To plan for workload management and monitoring, consider the following factors:

- Workload monitoring in dedicated Azure Monitor Logs workspaces.

For workloads that are deployed to virtual machines (VMs), store logs relative to dedicated Azure Monitor Logs workspaces. Workload team members can access logs for their workloads or VMs according to their Azure role-based access control (RBAC) roles.
- Sovereign workloads that drive the use of dedicated Azure Monitor Logs workspaces.

For sovereign workloads that require customer-managed keys to encrypt data, you can provide a high level of security and control. Data is encrypted twice. Microsoft-managed or customer-managed keys encrypt data at the service level. Two encryption algorithms and two keys encrypt data at the infrastructure level. For more information, see [Dedicated clusters](#).
- Performance and health monitoring for infrastructure as a service (IaaS) and platform as a service (PaaS) resources. Data is encrypted twice: once at the service level by using Microsoft-managed keys or customer-managed keys, and once at the infrastructure level by using two different encryption keys and algorithms.
- Data aggregation across all workload components.
- Health modeling and operationalization:
 - How to measure the health of the workload and its subsystems.
 - A traffic-light model to represent health.
 - How to respond to failures across workload components.

For more information, see [Monitoring in a cloud environment](#).

Workload management and monitoring recommendations

You can use centralized Azure Monitor components to manage and monitor workloads:

- Use a centralized Azure Monitor Logs workspace to collect logs and metrics from IaaS and PaaS workload resources.
- Control workspace and log access with Azure RBAC. For more information, see [Azure Monitor access control overview](#).
- Use [Azure Monitor Metrics](#) for time-sensitive analysis.

Azure Monitor stores metrics in a time-series database optimized to analyze time-stamped data. Metrics are well suited for alerts and detecting issues quickly.

Metrics can also monitor system performance. You can combine metrics with logs to identify the root causes of issues.

- Use [Azure Monitor Logs](#) for insights and reporting.

Logs contain different types of data organized into records with different sets of properties. Logs are useful for analyzing complex data from a range of sources, such as performance data, events, and traces. If necessary, use shared storage accounts in the landing zone for Azure diagnostic extension log storage.

- Use [Azure Monitor alerts](#) for generating operational alerts. Azure Monitor alerts unify metric and log alerts, and use features like actions and smart groups for advanced management and remediation.

For more workload management considerations and recommendations, see [Operational excellence](#) in the Azure Well-Architected Framework.

Feedback

Was this page helpful?



Design area: Azure governance

Article • 05/14/2024

Use Azure governance to establish the tooling that you need to support cloud governance, compliance auditing, and automated guardrails.

Design area review

Roles or functions: Azure governance originates from [cloud governance](#). You might need to implement the [cloud platform](#) or a [cloud center of excellence](#) to define and apply certain technical requirements. Governance focuses on enforcing operations and security requirements, which might require [cloud security](#), [central IT](#), or [cloud operations](#).

Scope: Consider your decisions from [identity](#), [network](#), [security](#), and [management](#) design area reviews. Your team can compare review decisions from automated governance, which is part of the Azure landing zone accelerator. Review decisions can help you determine what to audit or enforce and what policies to automatically deploy.

Out of scope: Azure governance establishes the foundation for networking. But it doesn't address compliance-related components, such as advanced network security or automated guardrails to enforce networking decisions. You can address these networking decisions when you review compliance design areas that are related to [security](#) and [governance](#). The cloud platform team should address initial networking requirements before addressing more complex components.

New (greenfield) cloud environment: To start your cloud journey, [create a small set of subscriptions](#). You can use Bicep deployment templates to create your new Azure landing zones. For more information, see [Azure landing zones Bicep—Deployment flow](#).

Existing (brownfield) cloud environment: If you want to apply proven-practice Azure governance principles to existing Azure environments, consider the following guidance:

- Establish a [management baseline](#) for your hybrid or multicloud environment.
- Implement [Microsoft Cost Management](#) features, like billing scopes, budgets, and alerts, to ensure that you don't exceed your expense limit.
- Use [Azure Policy](#) to enforce governance guardrails on Azure deployments and trigger remediation tasks to bring existing Azure resources into a compliant state.

- Consider using [the Microsoft Entra entitlement management feature](#) to automate Azure access request workflows, access assignments, reviews, and expiration.
- Use [Azure Advisor](#) recommendations to ensure cost optimization and operational excellence in Azure, both of which are core principles of the [Microsoft Azure Well-Architected Framework](#).

The [Azure landing zones Bicep—Deployment flow](#) repository contains Bicep deployment templates that can accelerate your greenfield and brownfield Azure landing zone deployments. These templates have integrated Microsoft proven-practice governance guidance.

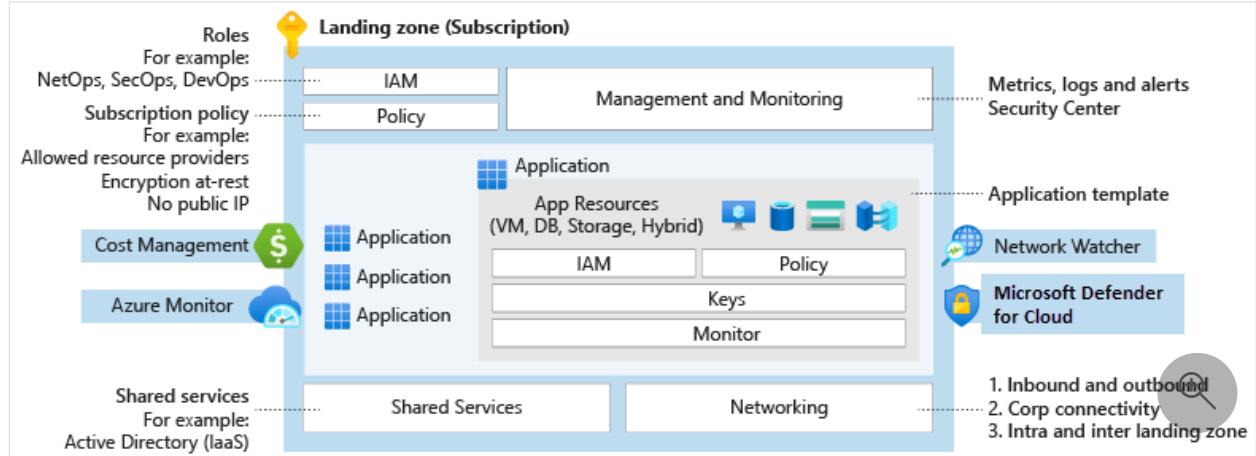
Consider using the [Azure landing zone default policy assignments](#) Bicep module to get a head start on ensuring compliance for your Azure environments.

For more information, see [Brownfield environment considerations](#).

Design area overview

Your organization's cloud adoption journey starts with strong controls for government environments.

Governance provides mechanisms and processes for maintaining control over platforms, applications, and resources in Azure.



Explore the following considerations and recommendations to make informed decisions as you plan your landing zone.

The governance design area focuses on design decisions for your landing zone. For information about governance processes and tools, see [Govern in the Cloud Adoption Framework for Azure](#).

Azure governance considerations

Azure Policy helps ensure security and compliance for enterprise technical estates. Azure Policy can enforce vital management and security conventions across Azure platform services. Azure Policy supplements Azure role-based access control (RBAC), which controls actions for authorized users. Cost Management can also help support your ongoing governance cost and spending in Azure or other multicloud environments.

Deployment considerations

Change advisory review boards can hinder your organization's innovation and business agility. Azure Policy replaces such reviews with automated guardrails and adherence audits to improve workload efficiency.

- Determine which Azure policies you need based on your business controls or compliance regulations. Use the policies included in the Azure landing zone accelerator as a starting point.
- Use the [policies included in Azure landing zones reference implementation](#) to consider other policies that might align to your business requirements.
- Enforce automated networking, identity, management, and security conventions.
- Manage and create policy assignments by using policy definitions that you can reuse at multiple inherited assignment scopes. You can have centralized baseline policy assignments at the management, subscription, and resource group scope.
- Ensure continuous compliance with compliance reporting and auditing.
- Understand that Azure Policy has limits, such as the restriction of definitions at any particular scope. For more information, see [Policy limits](#).
- Understand regulatory compliance policies. The policies might include HIPAA, PCI-DSS, or SOC 2 Trust Services Criteria.

Cost management considerations

- Consider the structure of your organization's cost and recharging model. Determine the key data points that accurately convey your cloud services spend.
- Choose the structure of tags that fits your cost and recharging model to help track your cloud spend.

- Use the Azure pricing calculator to estimate the expected monthly costs for using Azure products.
- Get Azure Hybrid Benefit to help reduce the cost of running your workloads in the cloud. You can use your on-premises Software Assurance-enabled Windows Server and SQL Server licenses on Azure. You can also use Red Hat and SUSE Linux subscriptions.
- Get Azure reservations and commit to one-year or three-year plans for multiple products. Reservation plans provide resource discounts, which can significantly reduce your resource costs by up to 72% compared to pay-as-you-go prices.
- Get the [Azure savings plan for compute](#) to save up to 65% compared to pay-as-you-go prices. Pick a one-year or three-year commitment that applies to compute services, regardless of your region, instance size, or operating system. Pick a plan for compute components, like virtual machines, dedicated hosts, container instances, Azure premium functions, and Azure app services. Combine an Azure savings plan with Azure reservations to optimize compute cost and flexibility.
- Use Azure policies to allow specific regions, resource types, and resource SKUs.
- Use the rule-based policy of Azure Storage lifecycle management to move blob data to the appropriate access tiers or to expire data at the end of the data lifecycle.
- Use Azure dev/test subscriptions to get a discount on access to select Azure services for nonproduction workloads.
- Use automatic scaling to dynamically allocate and deallocate resources to match your performance needs, which saves money.
- Use Azure Spot Virtual Machines to take advantage of unused compute capacity at a low cost. Spot Virtual Machines is great for workloads that can handle interruptions, for example batch-processing jobs, dev/test environments, and large-compute workloads.
- Select the right Azure services to help reduce costs. Some Azure services are free for 12 months and some are always free.
- Select the right compute service for your application to help improve cost efficiency. Azure offers many ways to host your code.

Resource management considerations

- Determine if the groups of resources in your environment can share required configurations, a common lifecycle, or common access constraints (such as RBAC) to help provide consistency.
- Choose an application or workload subscription design that's appropriate for your operation needs.
- Use standard resource configurations within your organization to ensure a consistent baseline configuration.

Security considerations

- Enforce tools and guardrails across the environment as part of a security baseline.
- Notify the appropriate people when you find deviations.
- Consider using Azure Policy to enforce tools, such as Microsoft Defender for Cloud, or guardrails, such as the Microsoft cloud security benchmark.

Identity management considerations

- Determine who has access to audit logs for identity and access management.
- Notify the appropriate people when suspicious sign-in events occur.
- Consider using [Microsoft Entra reports](#) to govern activity.
- Consider sending Microsoft Entra ID logs to the central Azure Monitor Logs workspace for the platform.
- Explore Microsoft Entra ID Governance features, like [access reviews](#) and [entitlement management](#).

Non-Microsoft tooling

- Use [AzAdvertiser](#) to get Azure governance updates. For example, you can find insights about policy definitions, initiatives, aliases, security, and regulatory compliance controls in Azure Policy or Azure RBAC role definitions. You can also get insight into resource provider operations, Microsoft Entra role definitions and role actions, and first-party API permissions.
- Use [Azure Governance Visualizer](#) to keep track of your technical governance estate. You can use the policy version checker feature for Azure landing zones to keep your environment up to date with the latest Azure landing zone policy release state.

Azure governance recommendations

Deployment acceleration recommendations

- Identify required Azure tags and use the append policy mode to enforce usage.
For more information, see [Define your tagging strategy](#).
- Map regulatory and compliance requirements to Azure Policy definitions and Azure role assignments.
- Establish Azure Policy definitions at the top-level root management group because they might be assigned at inherited scopes.
- Manage policy assignments at the highest appropriate level with exclusions at bottom levels, if necessary.
- Use Azure Policy to control resource provider registrations at the subscription or management group levels.
- Use built-in policies to minimize operational overhead.
- Assign the built-in Resource Policy Contributor role at a specific scope to enable application-level governance.
- Limit the number of Azure Policy assignments at the root management group scope to avoid managing exclusions at inherited scopes.

Cost management recommendations

- Use Cost Management to implement financial oversight on resources in your environment.
- Use tags, such as the cost center or project name, to append the resource metadata. This approach helps enable granular analysis of expenses.

Azure governance in the Azure landing zone accelerator

The Azure landing zone accelerator provides organizations with mature governance controls.

For example, you can implement:

- A management group hierarchy that groups resources by function or workload type. This approach encourages resource consistency.
 - A rich set of Azure policies that enables governance controls at the management group level. This approach helps verify that all resources are in scope.
-

Feedback

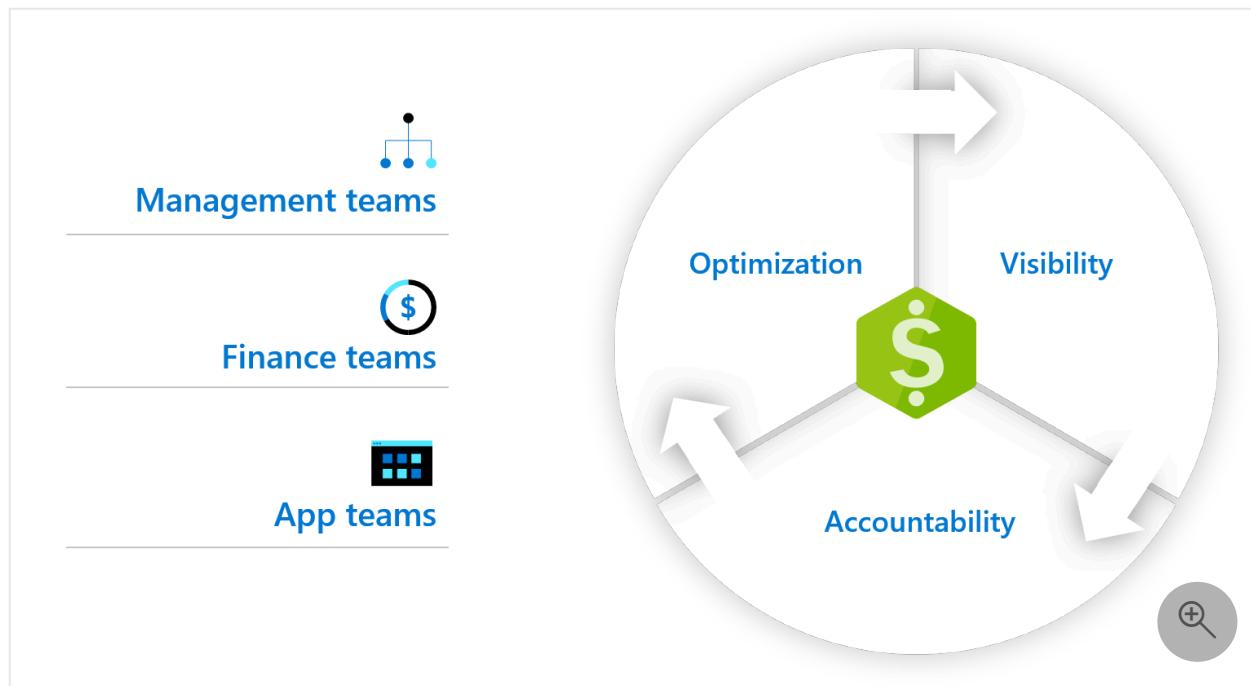
Was this page helpful?



Track costs across business units, environments, or projects

Article • 05/14/2024

To build a [cost-conscious organization](#), you need visibility and properly defined access to cost-related data. This best-practice article outlines decisions and implementation approaches to help you create tracking mechanisms to monitor costs. You'll learn how to apply fundamental Azure concepts to provide cost visibility.



Establish a well-managed environment

Cost control, much like governance and other management constructs, depends on a well-managed environment. To establish such an environment, especially a complex one, you need to consistently classify and organize all assets. Azure provides several mechanisms for classifying and organizing assets.

Assets, which are also known as resources, include all virtual machines, data sources, and applications deployed to the cloud. [Organize and manage your subscriptions](#) based on multiple criteria to establish a well-managed environment.

Classify assets

Tagging is an easy way to classify assets. Tagging associates metadata to an asset. That metadata can be used to classify the asset based on various data points. Tagging is a

fundamental part of any well-managed environment, and it's necessary for establishing proper governance of any environment.

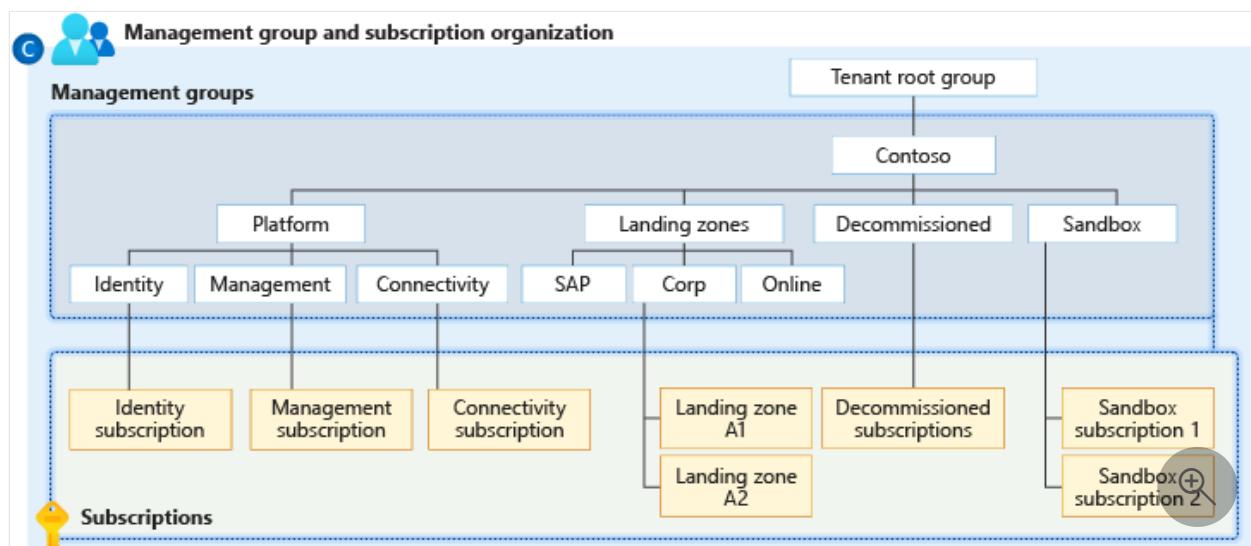
The first step is to [develop naming and tagging standards](#). The second step is to ensure that the tagging standard is consistently applied by [establishing a governance MVP](#).

When tags are used to classify assets as part of a cost management effort, companies often need the following tags: business unit, department, billing code, geography, environment, project, and workload or application categorization. [Microsoft Cost Management](#), a tool for setting budgets and gaining visibility into cloud costs for Azure or AWS, can use these tags to create different views of cost data.

Organize assets

There are several approaches to organizing assets. Microsoft's enterprise-scale [Azure landing zone](#) design provides an architecture that can be used as the basis of any Azure cloud environment. The landing zone [resource organization](#) documentation provides detailed guidance on organizing [management groups](#) and [subscriptions](#). Understanding the [design principles](#) used in designing the conceptual architecture will give you a foundation in best practices as you adapt the architecture to meet your specific business needs. Deviations to the design may be necessary to meet your business requirements, but understanding the impact of those deviations will prepare you for any necessary mitigations.

The following model for management groups, subscriptions, and resource groups creates a hierarchy that provides each team with the right level of visibility to perform their duties. When the enterprise needs cost controls to prevent budget overrun, it can apply governance tooling like Azure Policy to the subscriptions within this structure to quickly block future cost errors.



The rest of this article assumes the use of the best-practice approach in the preceding diagram. But the following articles can help you apply the approach to a resource organization that best fits your company:

- [Scale your Azure environment with multiple subscriptions](#)
- [Organize and manage your Azure subscriptions](#)
- [Deploy a governance MVP to govern well-managed environment standards](#)

Provide the right level of cost access

Managing cost is a team activity. The organization readiness section of the Cloud Adoption Framework defines a few core teams and outlines how those teams support cloud adoption efforts.

For the proper level of visibility into cost management data, the members of the team are assigned scope and roles. **Roles** define what a user can do to various assets. The **Scope** defines which assets, such as user, group, service principal, or managed identity, that a user can affect. As a general best practice, we suggest a least-privilege model in assigning people to various roles and scopes.

Roles

Cost Management supports the following built-in roles for each scope:

- **Owner:** Can view costs and manage everything, including cost configuration.
- **Contributor:** Can view costs and manage everything, including cost configuration, but excluding access control.
- **Reader:** Can view everything, including cost data and configuration, but can't make changes.
- [**Cost Management Contributor:**](#) Can view costs and manage cost configuration.
- [**Cost Management Reader:**](#) Can view cost data and configuration.

As a general best practice, members of all teams should be assigned the role of Cost Management Contributor. This role grants access to create and manage budgets to more effectively monitor and report on costs. But members of the [cloud strategy team](#) should be set to Cost Management Reader only, because they're not involved in setting budgets within the Cost Management tool.

Scope

The following scope and role settings will create the required visibility into cost management. This best practice might require minor changes to align to asset

organization decisions.

- **Cloud adoption team.** As cloud adoption teams primarily focus on implementation of cloud technologies, cost management access to production environments is not typically required. By virtue of normally having Contributor access to non-production or Sandbox subscriptions, this team would inherently have access to cost management data for those subscriptions.
- **Cloud strategy team.** Responsibilities for tracking costs across multiple projects and business units require [Cost Management Reader](#) access at the root level of the management group hierarchy.
 - Assign Cost Management Reader access to this team at the management group, which ensures ongoing visibility into all deployments associated with the subscriptions governed by that management group hierarchy.
- **Cloud governance team.** Responsibilities for managing cost, budget alignment, and reporting across all adoption efforts requires [Cost Management Contributor](#) access at the root level of the management group hierarchy.
 - In a well-managed environment, the cloud governance team likely has a higher degree of access already, making additional scope assignment for Cost Management Contributor unnecessary.
- **Cloud center of excellence.** Responsibility for managing costs related to shared services requires [Cost Management Contributor](#) access at the subscription level. Additionally, this team might require Cost Management Contributor access to resource groups or subscriptions that contain assets deployed by CCoE automation processes to understand how those processes affect costs.
 - **Shared services.** When a cloud center of excellence is engaged, best practice suggests that assets managed by the CCoE are supported from a centralized shared service subscription within a hub and spoke model. In this scenario, the CCoE likely has Contributor or Owner access to that subscription, making additional scope assignment for Cost Management Contributor unnecessary.
 - **CCoE automation/controls.** The CCoE commonly provides controls and automated deployment scripts to cloud adoption teams. The CCoE has a responsibility to understand how these accelerators affect costs. To gain that visibility, the team needs Cost Management Contributor access to any resource groups or subscriptions running those accelerators.
- **Cloud operations team.** Responsibility for managing ongoing costs of production environments requires [Cost Management Contributor](#) access to the Landing Zone and Platform management group nodes.

- The general recommendation puts production and nonproduction assets in separate subscriptions that are governed by nodes of the management group hierarchy associated with production environments. In a well-managed environment, members of the operations team likely have Owner or Contributor access to production subscriptions already, making the Cost Management Contributor role unnecessary.

Extra cost management resources

After you establish access to a well-managed environment hierarchy, the following articles can help you use that tool to monitor and control costs.

Use Cost Management

- Create and manage budgets
- Export cost data
- Optimize costs based on recommendations
- Use cost alerts to monitor usage and spending

Use Cost Management to govern AWS costs

- Set up AWS Cost and Usage Reports integration
- Manage AWS costs

Establish access, roles, and scope

- Understanding cost management scope
- Setting scope for a resource group

Next steps

To get started with Cost Management, see [How to optimize your cloud investment with Cost Management](#).

Feedback

Was this page helpful?

 Yes

 No

How to optimize your cloud investment with Cost Management

Article • 03/21/2024

Cost Management gives you the tools to plan for, analyze and reduce your spending to maximize your cloud investment. This document provides you with a methodical approach to cost management and highlights the tools available to you as you address your organization's cost challenges. Azure makes it easy to build and deploy cloud solutions. However, it's important that those solutions are optimized to minimize the cost to your organization. Following the principles outlined in this document and using our tools will help to make sure your organization is prepared for success.

Methodology

Cost management is an organizational problem and should be an ongoing practice that begins before you spend money on cloud resources. To successfully implement cost management and optimize costs, your organization must:

- Be prepared with the proper tools for success
- Be accountable for costs
- Take appropriate action to optimize spending

Three key groups, outlined below, must be aligned in your organization to make sure that you successfully manage costs.

- **Finance** - People responsible for approving budget requests across the organization based on cloud spending forecasts. They pay the corresponding bill and assign costs to various teams to drive accountability.
- **Managers** - Business decision makers in an organization that need to understand cloud spending to find the best spending results.
- **App teams** - Engineers managing cloud resources on a day-to-day basis, developing services to meet the organization's needs. These teams need the flexibility to deliver the most value in their defined budgets.

Key principles

Use the principles outlined below to position your organization for success in cloud cost management.

To learn more, watch the [Cost Management setting up for success](#) video. To watch other videos, visit the [Cost Management YouTube channel](#).
<https://www.youtube-nocookie.com/embed/dVuwITdSAZ4>

Planning

Comprehensive, up-front planning allows you to tailor cloud usage to your specific business requirements. Ask yourself:

- What business problem am I solving?
- What usage patterns do I expect from my resources?

Your answers will help you select the offerings that are right for you. They determine the infrastructure to use and how it's used to maximize your Azure efficiency.

Visibility

When structured well, Cost Management helps you to inform people about the Azure costs they're responsible for or for the money they spend. Azure has services designed to give you insight into *where* your money is spent. Take advantage of these tools. They can help you find resources that are underused, remove waste, and maximize cost-saving opportunities.

Accountability

Attribute costs in your organization to make sure that people responsible are accountable for their team's spending. To fully understand your organization's Azure spending, you should organize your resources to maximize insight into cost attribution. Good organization helps to manage and reduce costs and hold people accountable for efficient spending in your organization.

Optimization

Act to reduce your spending. Make the most of it based on the findings gathered through planning and increasing cost visibility. You might consider purchase and licensing optimizations along with infrastructure deployment changes that are discussed in detail later in this document.

Iteration

Everyone in your organization must engage in the cost management lifecycle. They need to stay involved on an ongoing basis to optimize costs. Be rigorous about this iterative process and make it a key tenet of responsible cloud governance in your organization.



Plan with cost in mind

Before you deploy cloud resources, assess the following items:

- The Azure offer that best meets your needs
- The resources you plan to use
- How much they might cost

Azure provides tools to assist you in the assessment process. The tools can give you a good idea of the investment required to enable your workloads. Then you can select the best configuration for your situation.

Azure onboarding options

The first step in maximizing your experience within Cost Management is to investigate and decide which Azure offer is best for you. Think about how you plan to use Azure in the future. Also consider how you want your billing model configured. Consider the following questions when making your decision:

- How long do I plan to use Azure? Am I testing, or do I plan to build longer-term infrastructure?
- How do I want to pay for Azure? Should I prepay for a reduced price or get invoiced at the end of the month?

To learn more about the various options, visit [How to buy Azure](#). Several of the most common billing models are identified below.

Free ↗

- 12 months of popular free services
- \$200 credit in your billing currency to explore services for 30 days
- 25+ services are always free

Pay as you go ↗

- No minimums or commitments
- Competitive Pricing
- Pay only for what you use
- Cancel anytime

Enterprise Agreement ↗

- Options for up-front Azure Prepayment (previously called monetary commitment)
- Access to reduced Azure pricing

Azure in CSP ↗

- CSP partners are the first point of contact for their customers' needs and the center of the customer relationship
- CSP partners provision new customers, order subscriptions, manage subscriptions, and perform admin tasks on behalf of their customers
- CSP partners bundle services with unique solutions or resell Azure while controlling the pricing, terms and billing

Estimate the cost of your solution

Before you deploy any infrastructure, assess how much your solution will cost. The assessment will help you create a budget for your organization for the workload, up-front. Then you can use a budget over time to benchmark the validity of your initial estimation. And you can compare it with the actual cost of your deployed solution.

Azure pricing calculator

The Azure pricing calculator allows you to mix and match different combinations of Azure services to see an estimate of the costs. You can implement your solution using different ways in Azure - each might influence your overall spending. Thinking early about all of the infrastructure needs of your cloud deployment helps you use the tool most effectively. It can help you get a solid estimate of your estimated spending in Azure.

For more information, see the [Azure pricing calculator](#).

Azure Migrate

Azure Migrate is a service that assesses your organization's current workloads in on-premises datacenters. It gives you insight into what you might need from an Azure replacement solution. First, Migrate analyzes your on-premises machines to determine whether migration is feasible. Then, it recommends VM sizing in Azure to maximize performance. Finally, it also creates a cost estimate for an Azure-based solution.

For more information, see [Azure Migrate](#).

Analyze and manage your costs

Keep informed about how your organization's costs evolve over time. Use the following techniques to properly understand and manage your spending.

Organize resources to maximize cost insights and accountability

A well-planned organizational structure for your Azure billing and resource hierarchies helps to give you a good understanding and control over costs as you create your cloud infrastructure. Watch the video [Setting up entity hierarchies](#) to gain a better understanding of the organizational tools that are available and how to take advantage of them. To watch other videos, visit the [Cost Management YouTube channel](#).

<https://www.youtube-nocookie.com/embed/n3TLRaYJ1NY>

As you evaluate and create a hierarchy that meets your needs, ask yourself the following questions.

Which billing hierarchy is available to me and what are the different scopes that I can use?

Identify the billing arrangement for your organization by determining your Azure offer type. The available scopes for each Azure billing arrangement are documented at [Understand and work with scopes](#).

If I have multiple teams, how should I organize my subscriptions and resource groups?

Creating a subscription or resource group for each team is a common practice. They can help you to differentiate costs and hold teams accountable. However, costs are bound to the subscription or resource group.

If you already have teams with multiple subscriptions, consider grouping the subscriptions into management groups to analyze the costs together. Management groups, subscriptions, and resource groups are all part of the Azure RBAC hierarchy. Use them collectively for access control in your teams.

Resources can span across multiple scopes, especially when they're shared by multiple teams or workloads. Consider identifying resources with tags. Tags are discussed further in the next section.

Do I have Development and Production environments?

Consider creating Dev/Test subscriptions for your development environments to take advantage of reduced pricing. If the workloads span multiple teams or Azure scopes, consider using tags to identify them.

Tag shared resources

Tags are an effective way to understand costs that span across multiple teams and Azure scopes. For example, you might have a resource like an email server that many teams use. You can put a shared resource, like the email server, in a subscription that's dedicated to shared resources or put it in an existing subscription. If you put it in an existing subscription, the subscription owner might not want its cost accruing to their team every month. For this example, you can use a tag to identify the resource as being shared.

Similarly, you might also have web apps or environments, such as Test or Production, that use resources across multiple subscriptions owned by different teams. To better understand the full cost of the workloads, tag the resources that they use. When tags are applied properly, you can apply them as a filter in cost analysis to better understand trends.

After you plan for resource tagging, you can configure an Azure Policy definition to enforce tagging on resources. Watch the [How to review tag policies with Cost Management](#) video to understand the tools available that help you enforce scalable resource tagging. To watch other videos, visit the [Cost Management YouTube channel](#).
<https://www.youtube-nocookie.com/embed/nHQYcYGKuyw>

Use cost analysis

Cost analysis allows you to analyze your organizational costs in-depth by slicing and dicing your costs using standard resource properties. Consider the following common

questions as a guide for your analysis. Answering these questions on a regular basis will help you stay more informed and enable more cost-conscious decisions.

- **Estimated costs for the current month** – How much have I incurred so far this month? Will I stay under my budget?
- **Investigate anomalies** – Do routine checks to make sure that costs stay within a reasonable range of normal usage. What are the trends? Are there any outliers?
- **Invoice reconciliation** – Is my latest invoiced cost more than the previous month? How did spending habits change month-over-month?
- **Internal chargeback** – Now that I know how much I'm being charged, how should those charges be broken down for my organization?

For more information, see [cost analysis](#).

Export billing data on a schedule

Do you need to import your billing data into an external system, like a dashboard or financial system? Set up automated exports to Azure Storage and avoid manually downloading files every month. You can then easily set up automatic integrations with other systems to keep your billing data in sync.

For more information about exporting billing data, see [Create and manage exported data](#).

Create budgets

After you've identified and analyzed your spending patterns, it's important to begin setting limits for yourself and your teams. Budgets give you the ability to set either a cost or usage-based budget with many thresholds and alerts. Make sure to review the budgets that you create regularly to see your budget burn-down progress and make changes as needed. Budgets also allow you to configure an automation trigger when a given budget threshold is reached. For example, you can configure your service to shut down VMs. Or you can move your infrastructure to a different pricing tier in response to a budget trigger.

For more information, see [Create budgets](#).

For more information about budget-based automation, see [Budget Based Automation](#).

Act to optimize

Use the following ways to optimize spending.

Cut out waste

After you've deployed your infrastructure in Azure, it's important to make sure it is being used. The easiest way to start saving immediately is to review your resources and remove any that aren't being used. From there, you should determine if your resources are being used as efficiently as possible.

Azure Advisor

Azure Advisor is a service that, among other things, identifies virtual machines with low utilization from a CPU or network usage standpoint. From there, you can decide to either shut down or resize the machine based on the estimated cost to continue running the machines. Advisor also provides recommendations for reserved instance purchases. The recommendations are based on your last 30 days of virtual machine usage. When acted on, the recommendations can help you reduce your spending.

For more information, see [Azure Advisor](#).

Size your VMs properly

VM sizing has a significant impact on your overall Azure cost. The number of VMs needed in Azure might not equate to what you currently have deployed in an on-premises datacenter. Make sure you choose the right size for the workloads that you plan to run.

For more information, see [Azure IaaS: proper sizing and cost](#).

Use purchase discounts

Azure has many discounts that your organization should take advantage of to save money.

Azure savings plan for compute

Azure savings plan for compute is our most flexible savings plan. It lets you save up to 65 percent on pay-as-you-go prices and applies to a broad range of compute services across subscriptions, resource groups, management groups or entire Azure accounts. You select an hourly compute commitment for a one-year or three-year term. The longer the commitment, the more savings you earn. You can pay monthly for no additional cost, and Azure automatically applies the largest savings to your account.

For more information, see [Azure savings plan for compute](#).

Azure Reservations

Azure Reservations allow you to prepay for one-year or three-years of virtual machine or SQL Database compute capacity. Pre-paying will allow you to get a discount on the resources you use. Azure reservations can significantly reduce your virtual machine or SQL database compute costs — up to 72 percent on pay-as-you-go prices with one-year or three-year upfront commitment. Reservations provide a billing discount and don't affect the runtime state of your virtual machines or SQL databases.

For more information, see [What are Azure Reservations?](#).

Use Azure Hybrid Benefit

If you already have Windows Server or SQL Server licenses in your on-premises deployments, you can use the Azure Hybrid Benefit program to save in Azure. With the Windows Server benefit, each license covers the cost of the OS (up to two virtual machines), and you only pay for base compute costs. You can use existing SQL Server licenses to save up to 55 percent on vCore-based SQL Database options. Options include SQL Server in Azure Virtual Machines and SQL Server Integration Services.

For more information, see [Azure Hybrid Benefit savings calculator](#).

Other resources

Azure also has a service that allows you to build services that take advantage of surplus capacity in Azure for reduced rates. For more information, see [Use low priority VMs with Batch](#).

Next steps

- If you're new to Cost Management, read [What is Cost Management?](#) to learn how it helps monitor and control Azure spending and to optimize resource use.

Tutorial: Create and manage budgets

Article • 07/09/2024

Budgets in Cost Management help you plan for and drive organizational accountability. They help you proactively inform others about their spending to manage costs and monitor how spending progresses over time.

You can configure alerts based on your actual cost or forecasted cost to ensure that your spending is within your organizational spending limit. Notifications are triggered when the budget thresholds are exceeded. Resources aren't affected, and your consumption isn't stopped. You can use budgets to compare and track spending as you analyze costs.

Cost and usage data is typically available within 8-24 hours and budgets are evaluated against these costs every 24 hours. Be sure to get familiar with [Cost and usage data updates](#) specifics. When a budget threshold is met, email notifications are normally sent within an hour of the evaluation.

Budgets reset automatically at the end of a period (monthly, quarterly, or annually) for the same budget amount when you select an expiration date in the future. Because they reset with the same budget amount, you need to create separate budgets when budgeted currency amounts differ for future periods. When a budget expires, it automatically gets deleted.

The examples in this tutorial walk you through creating and editing a budget for an Azure Enterprise Agreement (EA) subscription.

Watch the [Apply budgets to subscriptions using the Azure portal](#) video to see how you can create budgets in Azure to monitor spending. To watch other videos, visit the [Cost Management YouTube channel](#).

<https://www.youtube-nocookie.com/embed/UrkHiUx19Po>

In this tutorial, you learn how to:

- ✓ Create a budget in the Azure portal
- ✓ Create and edit budgets
- ✓ Create a budget with an Azure Resource Manager template

Prerequisites

Budgets are supported for the following types of Azure account types and scopes:

- Azure role-based access control (Azure RBAC) scopes
 - Management groups
 - Subscription
- Enterprise Agreement scopes
 - Billing account
 - Department
 - Enrollment account
- Individual agreements
 - Billing account
- Microsoft Customer Agreement scopes
 - Billing account - Budget evaluation only supports USD currency, not the billing currency. An exception is that customers in the China 21V cloud have their budgets evaluated in CNY currency.
 - Billing profile
 - Invoice section
 - Customer
- AWS scopes
 - External account
 - External subscription

 **Note**

The Connector for AWS in the Cost Management service retires on March 31, 2025. Users should consider alternative solutions for AWS cost management reporting. On March 31, 2024, Azure will disable the ability to add new Connectors for AWS for all customers. For more information, see [Retire your Amazon Web Services \(AWS\) connector](#).

To view budgets, you need at least read access for your Azure account.

If you have a new subscription, you can't immediately create a budget or use other Cost Management features. It might take up to 48 hours before you can use all Cost Management features.

Read access is required to view budgets for Azure EA subscriptions. To create and manage budgets, you must have contributor permission.

The following Azure permissions, or scopes, are supported per subscription for budgets by user and group.

- Owner – Can create, modify, or delete budgets for a subscription.

- Contributor and Cost Management contributor – Can create, modify, or delete their own budgets. Can modify the budget amount for budgets created by others.
- Reader and Cost Management reader – Can view budgets that they have permission to.

For more information about scopes, including access needed to configure exports for Enterprise Agreement and Microsoft Customer agreement scopes, see [Understand and work with scopes](#). For more information about assigning permission to Cost Management data, see [Assign access to Cost Management data](#).

Sign in to Azure

- Sign in to the [Azure portal](#).

Create a budget in the Azure portal

You can create an Azure subscription budget for a monthly, quarterly, or annual period.

To create or view a budget, open a scope in the Azure portal and select **Budgets** in the menu. For example, navigate to **Subscriptions**, select a subscription from the list, and then select **Budgets** in the menu. Use the **Scope** pill to switch to a different scope, like a management group, in Budgets. For more information about scopes, see [Understand and work with scopes](#).

If you want to create a budget for a resource group, ensure that you navigate to one first. You can navigate to a resource group by searching for **Resource groups** in the Azure portal search box. Then, select a resource group from the list. Afterward, the **Budgets** option is available in the menu.

After you create budgets, they show a simple view of your current spending against them.

Select **Add**.

Name	Scope	Reset period	Creation date	Expiration date	Budget	Forecasted	Evaluated spend	Progress
EAAccount-BoTest...	8608480 (Billing acc...)	Monthly	2/1/2021	1/31/2023	\$100.00	\$0.00	\$0.00	0.00%
EAAccount-BoTest-A...	208903 (Enrollment ...)	Monthly	2/1/2021	1/31/2023	\$100.00	\$0.00	\$0.00	0.00%
Pri_Forecast	8608480 (Billing acc...)	Monthly	2/1/2021	2/28/2022	\$100.00	\$120.5K	\$26,046	100.00%
Pri_actualForecast	8608480 (Billing acc...)	Monthly	2/1/2021	1/31/2023	\$200.00	\$120.5K	\$26,046	100.00%
EAAccount-BoTest-...	48420 (Department)	Monthly	2/1/2021	1/31/2023	\$100.00	\$4,572	\$1,088	100.00%
Pri_EdgecaseTest	8608480 (Billing acc...)	Monthly	2/1/2021	2/28/2022	\$100.00	\$515.23	\$131.43	100.00%
ACM_Department_B...	84820 (Department)	Monthly	10/1/2019	9/30/2021	\$55,000.00	...	\$4,272	7.77%
Enrollment_budget	8608480 (Billing acc...)	Monthly	4/1/2020	3/31/2022	\$45,000.00	...	\$26,046	57.88%
ACM	8608480 (Billing acc...)	Monthly	8/1/2020	7/31/2022	\$30,000.00	...	\$0.00	0.00%
JoTestBudget	8608480 (Billing acc...)	Monthly	11/1/2020	10/31/2022	\$50,000.00	...	\$26,046	52.09%
DemoTestBudget	8608480 (Billing acc...)	Monthly	12/1/2020	11/30/2022	\$40,000.00	...	\$26,046	65.12%
ClaroTGT	8608480 (Billing acc...)	Monthly	12/1/2020	11/30/2022	\$35,000.00	...	\$26,046	74.42%

In the **Create budget** window, make sure that the scope shown is correct. Choose any filters that you want to add. Filters allow you to create budgets on specific costs, such as resource groups in a subscription or a service like virtual machines. For more information about the common filter properties that you can use in budgets and cost analysis, see [Group and filter properties](#).

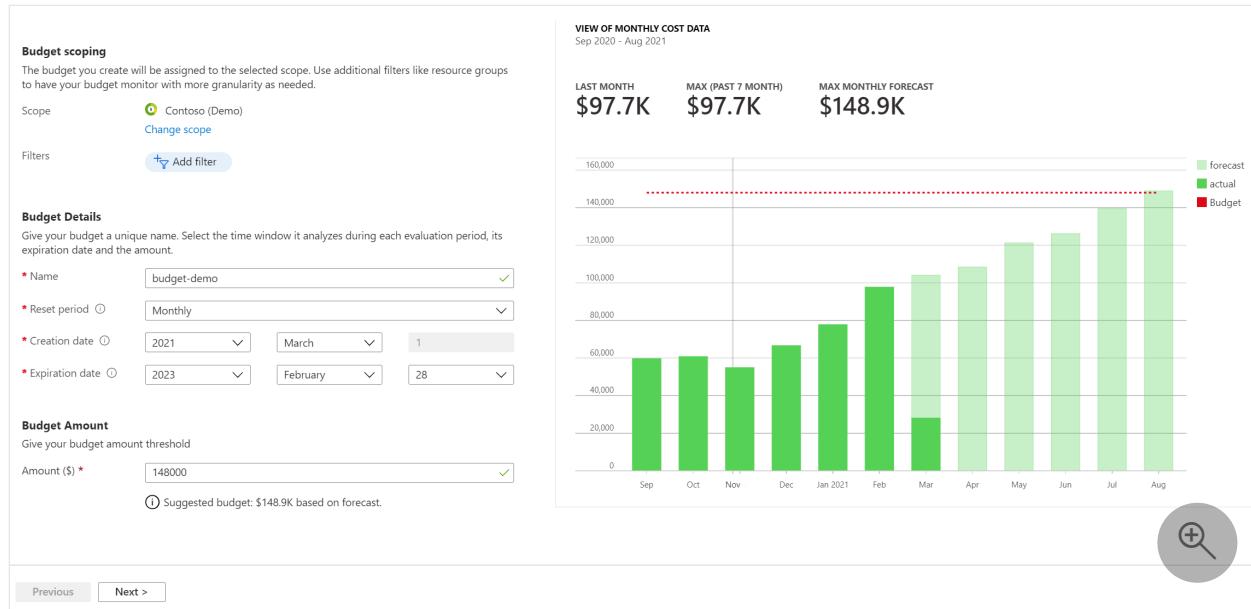
After you identify your scope and filters, type a budget name. Then, choose a monthly, quarterly, or annual budget reset period. The reset period determines the time window that gets analyzed by the budget. The cost evaluated by the budget starts at zero at the beginning of each new period. When you create a quarterly budget, it works in the same way as a monthly budget. The difference is that the budget amount for the quarter is evenly divided among the three months of the quarter. An annual budget amount is evenly divided among all 12 months of the calendar year.

If you have a pay-as-you-go, MSDN, or Visual Studio subscription, your invoice billing period might not align to the calendar month. For those subscription types and resource groups, you can create a budget aligned to your invoice period or to calendar months. To create a budget aligned to your invoice period, select a reset period of **Billing month**, **Billing quarter**, or **Billing year**. To create a budget aligned to the calendar month, select a reset period of **Monthly**, **Quarterly**, or **Annually**.

Next, identify the expiration date when the budget becomes invalid and stops evaluating your costs.

Based on the fields chosen in the budget so far, a graph is shown to help you select a threshold to use for your budget. The suggested budget is based on the highest

forecasted cost that you might incur in future periods. You can change the budget amount.



After you configure the budget amount, select **Next** to configure budget alerts for actual cost and forecasted budget alerts.

Configure actual costs budget alerts

Budgets require at least one cost threshold (% of budget) and a corresponding email address. You can optionally include up to five thresholds and five email addresses in a single budget. When a budget threshold is met, email notifications are normally sent within an hour of the evaluation. Actual costs budget alerts are generated for the actual cost accrued in relation to the budget thresholds configured.

Configure forecasted budget alerts

Forecasted alerts provide advanced notification that your spending trends are likely to exceed your budget. The alerts use forecasted cost predictions. Alerts are generated when the forecasted cost projection exceeds the set threshold. You can configure a forecasted threshold (% of budget). When a forecasted budget threshold is met, notifications are normally sent within an hour of the evaluation.

To toggle between configuring an Actual vs Forecasted cost alert, use the **Type** field when configuring the alert as shown in the following image.

If you want to receive emails, add azure-noreply@microsoft.com to your approved senders list so that emails don't go to your junk email folder. For more information about notifications, see [Use cost alerts](#).

In the following example, an email alert gets generated when 90% of the budget is reached. If you create a budget with the Budgets API, you can also assign roles to people to receive alerts. Assigning roles to people isn't supported in the Azure portal. For more about the Budgets API, see [Budgets API](#). If you want to have an email alert sent in a different language, see [Supported locales for budget alert emails](#).

Alert limits support a range of 0.01% to 1000% of the budget threshold.

✓ Create a budget ✓ Set alerts

Configure alert conditions and send email notifications based on your spend.

* Alert conditions

Type	% of budget	Amount
Actual	90	133200
Forecasted	100	148000
Select type	Enter %	-

* Alert recipients (email)

Alert recipients (email)

user@contoso.com
example@email.com

It is recommended to add azure-noreply@microsoft.com to your email white list to ensure alert mails do not go to your spam folder.

Language preference

Select your preferred language for receiving the alert email for all recipients provided above. Default is the language associated to your enrollment.

Languages * Default

1 Your budget evaluation will begin in a few hours. Learn more

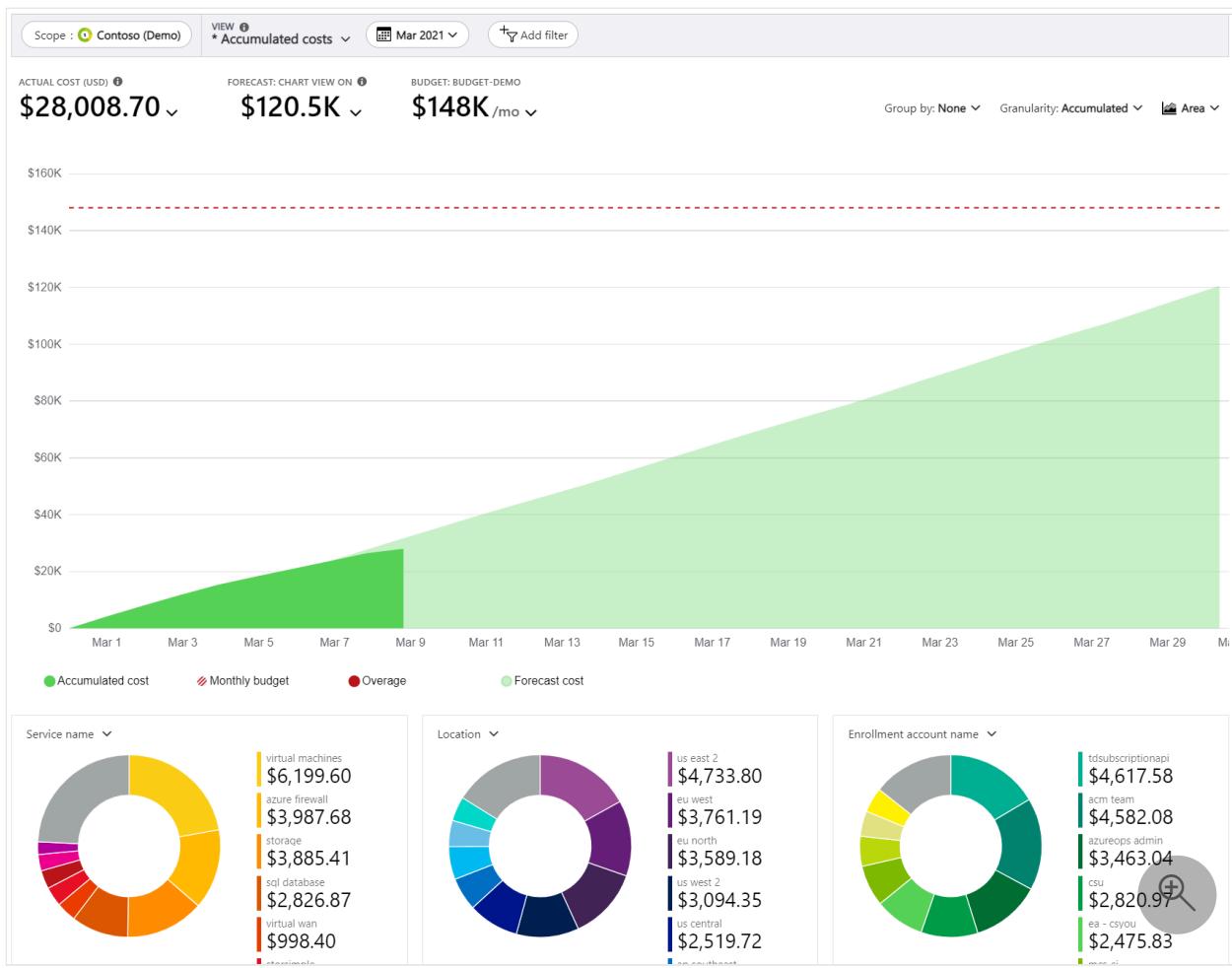
VIEW OF MONTHLY COST DATA
Sep 2020 - Aug 2021

LAST MONTH \$97.7K MAX (PAST 7 MONTH) \$97.7K MAX MONTHLY FORECAST \$148.9K

Previous Create

+

After you create a budget, it appears in cost analysis. Viewing your budget against your spending trend is one of the first steps when you start to [analyze your costs and spending](#).



In the preceding example, you created a budget for a subscription. You can also create a budget for a resource group. If you want to create a budget for a resource group, navigate to **Cost Management + Billing > Subscriptions >** select a subscription **> Resource groups >** select a resource group **> Budgets >** and then **Add a budget**.

Create a budget for combined Azure and AWS costs

You can group your Azure and AWS costs together by assigning a management group to your connector along with its consolidated and linked accounts. Assign your Azure subscriptions to the same management group. Then create a budget for the combined costs.

1. In Cost Management, select **Budgets**.
2. Select **Add**.
3. Select **Change scope** and then select the management group.
4. Continue creating the budget until complete.

Costs in budget evaluations

Budget cost evaluations now include reserved instance and purchase data. If the charges apply to you, then you might receive alerts as charges are incorporated into your

evaluations. Sign in to the [Azure portal](#) to verify that budget thresholds are properly configured to account for the new costs. Your Azure billed charges aren't changed. Budgets now evaluate against a more complete set of your costs. If the charges don't apply to you, then your budget behavior remains unchanged.

If you want to filter the new costs so that budgets are evaluated against first party Azure consumption charges only, add the following filters to your budget:

- Publisher Type: Azure
- Charge Type: Usage

Budget cost evaluations are based on actual cost. They don't include amortization. For more information about filtering options available to you in budgets, see [Understanding grouping and filtering options](#).

Trigger an action group

When you create or edit a budget for a subscription or resource group scope, you can configure it to call an action group. The action group can perform various actions when your budget threshold is met. You can receive mobile push notifications when your budget threshold is met by enabling [Azure app push notifications](#) while configuring the action group.

Action groups are currently only supported for subscription and resource group scopes. For more information about creating action groups, see [action groups](#).

For more information about using budget-based automation with action groups, see [Manage costs with budgets](#).

To create or update action groups, select **Manage action group** while you're creating or editing a budget.

✓ Create a budget

2 Set alerts

Configure alert conditions and send email notifications based on your spend.

* Alert conditions

Type	% of budget	Amount	Action group
Actual	90	799.2	None
Select type	Enter %	-	None

Manage action group ⓘ

Next, select **Add action group** and create the action group.

You can integrate budgets with action groups, regardless of whether the common alert schema is enabled or disabled in those groups. For more information on how to enable common alert schema, see [How do I enable the common alert schema?](#)

Budgets in the Azure mobile app

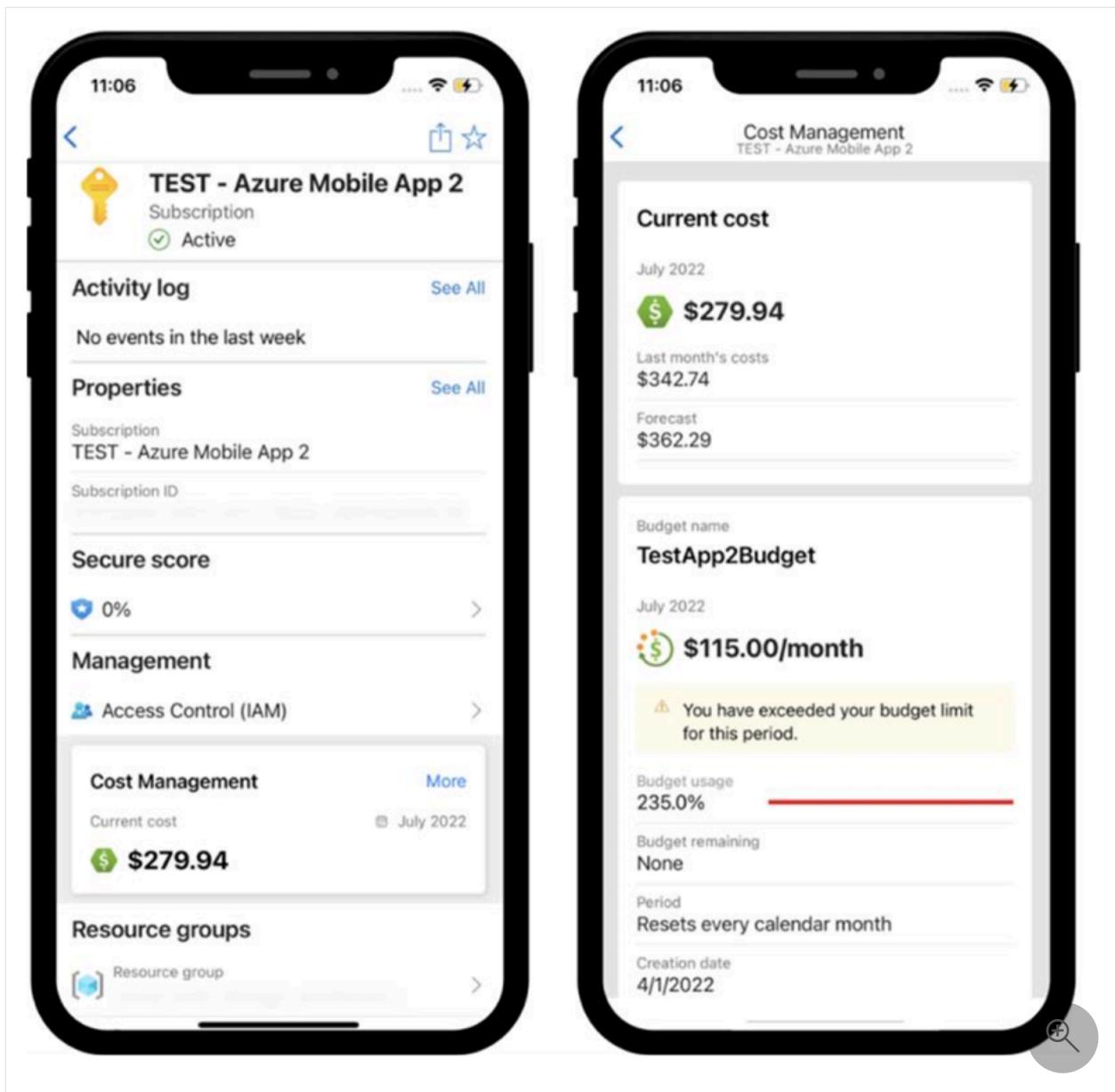
You can view budgets for your subscriptions and resource groups from the **Cost Management** card in the [Azure app](#).

1. Navigate to any subscription or resource group.
2. Find the **Cost Management** card and tap **More**.
3. Budgets load below the **Current cost** card. They're sorted by descending order of usage.

To receive mobile push notifications when your budget threshold is met, you can configure action groups. When setting up budget alerts, make sure to select an action group that has [Azure app push notifications](#) enabled.

ⓘ Note

Currently, the Azure mobile app only supports the subscription and resource group scopes for budgets.



Create and edit budgets

PowerShell

If you're an EA customer, you can create and edit budgets programmatically using the Azure PowerShell module. However, we recommend that you use REST APIs to create and edit budgets because CLI commands might not support the latest version of the APIs. Budgets created with PowerShell don't send notifications.

Note

Customers with a Microsoft Customer Agreement should use the [Budgets REST API](#) to create budgets programmatically.

To download the latest version of Azure PowerShell, run the following command:

```
Azure PowerShell
```

```
install-module -name Az
```

The following example commands create a budget using PowerShell. Make sure to replace all example prompts with your own info.

```
Azure PowerShell
```

```
#Sign into Azure PowerShell with your account  
  
Connect-AzAccount  
  
#Select a subscription to monitor with a budget  
  
select-AzSubscription -Subscription "Your Subscription"  
  
#Create an action group email receiver and corresponding action group  
  
$email1 = New-AzActionGroupReceiver -EmailAddress test@test.com -Name EmailReceiver1  
$ActionGroupId = (Set-AzActionGroup -ResourceGroupName YourResourceGroup -Name TestAG -ShortName TestAG -Receiver $email1).Id  
  
#Create a monthly budget that sends an email and triggers an Action Group to send a second email. Make sure the StartDate for your monthly budget is set to the first day of the current month. Note that Action Groups can also be used to trigger automation such as Azure Functions or Webhooks.  
  
Get-AzContext  
New-AzConsumptionBudget -Amount 100 -Name TestPSBudget -Category Cost -  
StartDate 2020-02-01 -TimeGrain Monthly -EndDate 2022-12-31 -  
ContactEmail test@test.com -NotificationKey Key1 -NotificationThreshold  
0.8 -NotificationEnabled -ContactGroup $ActionGroupId
```

Clean up resources

If you created a budget and you no longer need it, view its details and delete it.

Next steps

In this tutorial, you learned how to:

- ✓ Create a budget in the Azure portal
- ✓ Create and edit budgets with PowerShell
- ✓ Create a budget with an Azure Resource Manager template

Advance to the next tutorial to create a recurring export for your cost management data.

[Create and manage exported data](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#)

Tutorial: Create and manage exported data

Article • 08/14/2024

If you read the Cost Analysis tutorial, then you're familiar with manually downloading your Cost Management data. However, you can create a recurring task that automatically exports your Cost Management data to Azure storage on a daily, weekly, or monthly basis. Exported data is in CSV format and it contains all the information that Cost Management collects. You can then use the exported data in Azure storage with external systems and combine it with your own custom data. And you can use your exported data in an external system like a dashboard or other financial system.

Watch the [How to schedule exports to storage with Cost Management](#) video about creating a scheduled export of your Azure cost data to Azure Storage. To watch other videos, visit the [Cost Management YouTube channel](#).

https://www.youtube-nocookie.com/embed/rWa_xl1aRzo

The examples in this tutorial walk you through exporting your cost management data and then verify that the data was successfully exported.

In this tutorial, you learn how to:

- ✓ Create a daily export
- ✓ Verify that data is collected

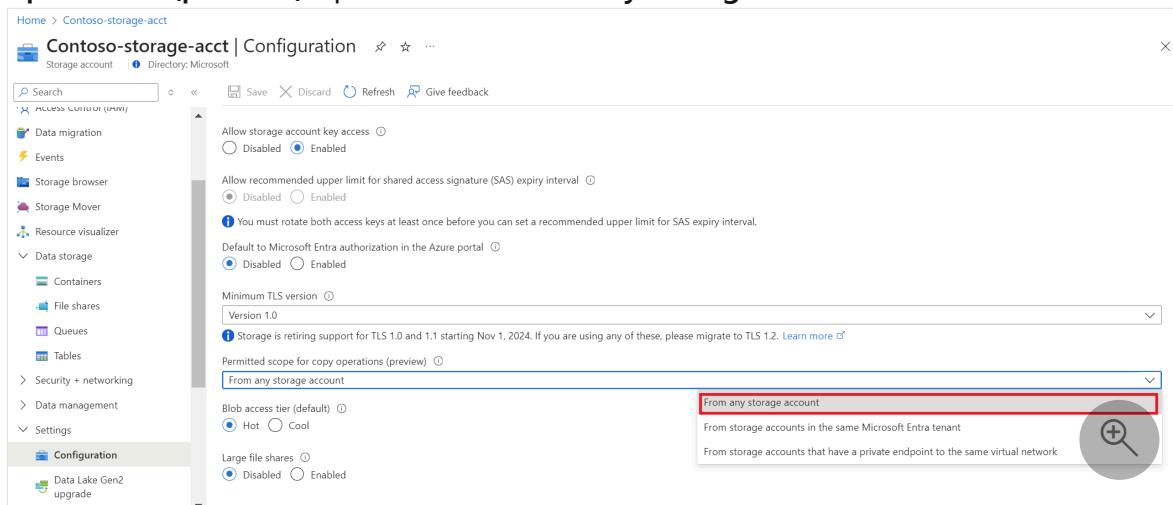
Prerequisites

Data export is available for various Azure account types, including [Enterprise Agreement \(EA\)](#) and [Microsoft Customer Agreement](#) customers. To view the full list of supported account types, see [Understand Cost Management data](#). The following Azure permissions, or scopes, are supported per subscription for data export by user and group. For more information about scopes, see [Understand and work with scopes](#).

- Owner - Can create, modify, or delete scheduled exports for a subscription.
- Contributor - Can create, modify, or delete their own scheduled exports. Can modify the name of scheduled exports created by others.
- Reader - Can schedule exports that they have permission to.
 - **For more information about scopes, including access needed to configure exports for Enterprise Agreement and Microsoft Customer agreement scopes, see [Understand and work with scopes](#).**

For Azure Storage accounts:

- Write permissions are required to change the configured storage account, independent of permissions on the export.
- Your Azure storage account must be configured for blob or file storage.
- Don't configure exports to a storage container when configured as a destination in an [object replication rule](#).
- To export to storage accounts with configured firewalls, you need other privileges on the storage account. The other privileges are only required during export creation or modification. They are:
 - Owner role on the storage account. Or
 - Any custom role with `Microsoft.Authorization/roleAssignments/write` and `Microsoft.Authorization/permissions/read` permissions. Additionally, ensure that you enable [Allow trusted Azure service access](#) to the storage account when you configure the firewall. If you want to use the [Exports REST API](#) to generate exports to a storage account located behind a firewall, use the API version 2023-08-01 or later version. All newer API versions continue to support exports behind the firewall.
- The storage account configuration must have the **Permitted scope for copy operations (preview)** option set to **From any storage account**.



If you have a new subscription, you can't immediately use Cost Management features. It might take up to 48 hours before you can use all Cost Management features.

Sign in to Azure

Sign in to the Azure portal at <https://portal.azure.com>.

Create a daily export

To create or view a data export or to schedule an export, choose a scope in the Azure portal and select **Cost analysis** in the menu. For example, navigate to **Subscriptions**, select a subscription from the list, and then select **Cost analysis** in the menu. At the top of the Cost analysis page, select **Configure subscription**, then **Exports**.

 **Note**

- Besides subscriptions, you can create exports on resource groups, management groups, departments, and enrollments. For more information about scopes, see [Understand and work with scopes](#).
- When you're signed in as a partner at the billing account scope or on a customer's tenant, you can export data to an Azure Storage account that's linked to your partner storage account. However, you must have an active subscription in your CSP tenant.

1. Select **Create**

2. For **Export details**, make a selection:

- Type a name for export
- **Daily export of month-to-date costs** - Provides a new export file daily for your month-to-date costs. The latest data is aggregated from previous daily exports.
- **Weekly export of cost for the last seven days** - Creates a weekly export of your costs for the past seven days from the selected start date of your export.
- **Monthly export of last month's costs** - Provides you with an export of your last month's costs compared to the current month that you create the export. Afterward, the schedule runs an export on the fifth day of every new month with your previous months costs.
- **One-time export** - Allows you to choose a date range for historical data to export to Azure blob storage. You can export a maximum of 90 days of historical costs from the day you choose. This export runs immediately and is available in your storage account within two hours. Depending on your export type, either choose a start date, or choose a **From** and **To** date.

3. Specify the subscription for your Azure storage account, then select a resource group or create a new one.
4. Select the storage account name or create a new one.
5. Select the location (Azure region).
6. Specify the storage container and the directory path that you'd like the export file to go to.

The screenshot shows the 'New export' configuration page in the Microsoft Azure portal. The 'Export details' section includes fields for Name (DemoExport), Metric (Actual cost (Usage and Purchases)), Export type (Daily export of month-to-date costs), and Start date (Wed Aug 05 2020). The 'Storage' section includes fields for Subscription (Trey Research Corporate), Resource group (TreyNetwork), Account name (cmdemo.core.windows.net), Location ((US) East US), Container (democontainer), and Directory (demodirectory). A 'Create' button is at the bottom left, and a search icon is at the bottom right.

7. Review your export details and select **Create**.

Your new export appears in the list of exports. By default, new exports are enabled. If you want to disable or delete a scheduled export, select any item in the list, and then select either **Disable** or **Delete**.

Initially, it can take 12-24 hours before the export runs. However, it can take up longer before data is shown in exported files.

Configure exports for storage accounts with a firewall

If you need to export to a storage account behind the firewall for security and compliance requirements, ensure that you have all [prerequisites](#) met.

ⓘ Note

If you have an existing scheduled export and your change your storage network configuration, you must update the export and save it to reflect the changes.

Enable **Allow trusted Azure services access** on the storage account. You can turn that on while configuring the firewall of the storage account, from the Networking page. Here's a screenshot showing the page.

The screenshot shows the 'Networking' page for a storage account named 'ContosoStorageAccount'. The left sidebar lists various settings like Data storage, Security + networking, Data management, and Monitoring. The 'Networking' section is selected. The main content area shows the 'Firewalls and virtual networks' tab. A note at the top states: 'Firewall settings restricting access to storage services will remain in effect for up to a minute after saving updated settings allowing access.' Below this, under 'Public network access', the 'Enabled from selected virtual networks and IP addresses' option is selected. Under 'Virtual networks', there is a table with columns: Virtual Network, Subnet, Address range, Endpoint Status, Resource Group, and Subscription. A note says 'No network selected.' Under 'Firewall', there is a section to add IP ranges with a note 'Add IP ranges to allow access from the internet or your on-premises networks.' A checkbox for 'Add your client IP address (98.97.35.51)' is shown. Under 'Resource instances', there are dropdowns for 'Resource type' and 'Instance name'. Under 'Exceptions', a checkbox for 'Allow Azure services on the trusted services list to access this storage account.' is checked and highlighted with a red box. Other options include 'Allow read access to storage logging from any network' and 'Allow read access to storage metrics from any network'. At the bottom, there is a 'Network Routing' section with options for 'Microsoft network routing' and 'Internet routing', and a note about publishing route-specific endpoints. A large circular button with a plus sign and a magnifying glass icon is in the bottom right corner.

If you missed enabling that setting, you can easily do so from the **Exports** page when creating a new export.

New export ...

Azure Big Data and Beyond Docs Team

Exports allow you to create a recurring task that automatically exports your Cost Management data to an Azure Blob Storage on a daily, weekly, or monthly basis. The exported data is in CSV format and contains all the cost and usage information collected by Cost Management. You will incur costs for the Azure storage. [Learn more](#)

Export details

Name *	<input type="text" value="contosoexport"/>
Metric *	<input type="text" value="Actual cost (Usage and Purchases)"/>
Export type *	<input type="text" value="Daily export of month-to-date costs"/>
Start date (UTC time) *	<input type="text" value="Fri Jul 14 2023 UTC"/>

File Partitioning

Enable partitioning if you have larger datasets and want your exports to be split into multiple files. Please note that if you have partitioning off and it is subsequently turned on, your file schema may change slightly. [Learn more](#)



Storage

Use existing Create new

Subscription *	<input type="text" value="Contoso"/>
Storage account *	<input type="text" value="contosostorageacct"/>
<input type="checkbox"/> Allow trusted Azure service access? * ⓘ <small>To export to this storage account, select the checkbox to allow access. Learn more</small>	
Container *	<input type="text" value="Container name"/>
Directory *	<input type="text" value="Directory path"/>

**Create**

A system-assigned managed identity is created for a new job export when created or modified. You must have permissions because Cost Management uses the privilege to assign the *StorageBlobDataContributor* role to the managed identity. The permission is restricted to the storage account container scope. After the export job is created or updated, the user doesn't require Owner permissions for regular runtime operations.

ⓘ Note

- When a user updates destination details or deletes an export, the *StorageBlobDataContributor* role assigned to the managed identity is automatically removed. To enable the system to remove the role assignment, the user must have `microsoft.Authorization/roleAssignments/delete` permissions. If the permissions aren't available, the user needs to manually remove the role assignment on the managed identity.
- Currently, firewalls are supported for storage accounts in the same tenant. However, firewalls on storage accounts aren't supported for cross-tenant exports.

Add exports to the list of trusted services. For more information, see [Trusted access based on a managed identity](#).

Export schedule

Scheduled exports get affected by the time and day of week of when you initially create the export. When you create a scheduled export, the export runs at the same frequency for each export that runs later. For example, the export runs once each UTC day for a daily export of month-to-date costs export set at a daily frequency. Similarly for a weekly export, the export runs every week on the same UTC day as it is scheduled.

Individual export runs can occur at different times throughout the day. So, avoid taking a firm dependency on the exact time of the export runs. Run timing depends on the active load present in Azure during a given UTC day. When an export run begins, your data should be available within 4 hours.

Exports are scheduled using Coordinated Universal Time (UTC). The Exports API always uses and displays UTC.

- When you create an export using the [Exports API](#), specify the `recurrencePeriod` in UTC time. The API doesn't convert your local time to UTC.
 - Example - A weekly export is scheduled on Friday, August 19 with `recurrencePeriod` set to 2:00 PM. The API receives the input as 2:00 PM UTC, Friday, August 19. The weekly export is scheduled to run every Friday.
- When you create an export in the Azure portal, its start date time is automatically converted to the equivalent UTC time.
 - Example - A weekly export is scheduled on Friday, August 19 with the local time of 2:00 AM IST (UTC+5:30) from the Azure portal. The API receives the input as 8:30 PM, Thursday, August 18. The weekly export is scheduled to run every Thursday.

Each export creates a new file, so older exports aren't overwritten.

Create an export for multiple subscriptions

You can use a management group to aggregate subscription cost information in a single container. Exports support management group scope for Enterprise Agreement but not for Microsoft Customer Agreement or other subscription types. Multiple currencies are also not supported in management group exports.

Exports at the management group scope support only usage charges. Purchases, including reservations and savings plans aren't supported. Amortized cost reports are also not supported. When you create an export from the Azure portal for a management

group scope, the metric field isn't shown because it defaults to the usage type. When you create a management group scope export using the REST API, choose **ExportType** as **Usage**.

1. Create one management group and assign subscriptions to it, if you haven't already.
2. In cost analysis, set the scope to your management group and select **Select this management group**.

The screenshot shows the Microsoft Azure portal with the URL https://portal.azure.com/?feature.arm_canary=true&feature.exportapiversion=2020-05-01-preview&.... The user is logged in as admin@contoso.com (CONTOSO). The main page displays 'Cost Management: Contoso (Demo) | Configuration'. On the left, the 'Cost Management' menu is open, showing 'Cost analysis' selected. The 'Scope' dropdown is set to 'Contoso (Demo)'. A modal window titled 'Select scope' is open, showing a list of scopes: 'Root management g...' (Trey Research), 'Trey US', 'Trey Worldwide', 'Cloudyn Software Ltd.', and '657473078308'. The button 'Select this management group' is highlighted with a red box.

3. Create an export at the scope to get cost management data for the subscriptions in the management group.

The screenshot shows the 'Exports' blade under 'Cost Management: Trey Research > Configuration'. At the top, there are buttons for '+ Add' (highlighted with a red box), 'Refresh', 'Run now', 'Enable', 'Disable', and 'Delete'. Below these are two input fields: 'Scope : [Trey Research]' and 'Search to filter items...'. An informational message says 'How satisfied are you with exports? →' with a feedback icon.

File partitioning for large datasets

If you have a Microsoft Customer Agreement, Microsoft Partner Agreement, or Enterprise Agreement, you can enable Exports to chunk your file into multiple smaller file partitions to help with data ingestion. When you initially configure your export, set the **File Partitioning** setting to **On**. The setting is **Off** by default.

File Partitioning

Enable partitioning if you have larger datasets and want your exports to be split into multiple files. Please note that if you have partitioning off and it is subsequently turned on, your file schema may change slightly. [Learn more](#)

On



If you don't have a Microsoft Customer Agreement, Microsoft Partner Agreement, or Enterprise Agreement, then you don't see the **File Partitioning** option.

Partitioning isn't currently supported for resource groups or management group scopes.

Update existing exports to use file partitioning

If you have existing exports and you want to set up file partitioning, create a new export. File partitioning is only available with the latest Exports version. There might be minor changes to some of the fields in the usage files that get created.

If you enable file partitioning on an existing export, you might see minor changes to the fields in file output. Any changes are due to updates that were made to Exports after you initially set yours up.

Partitioning output

When file partitioning is enabled, you get a file for each partition of data in the export along with a `_manifest.json` file. The manifest contains a summary of the full dataset and information for each file partition in it. Each file partition has headers and contains only a subset of the full dataset. To handle the full dataset, you must ingest each partition of the export.

Here's a `_manifest.json` example manifest file.

JSON

```
{  
  "manifestVersion": "2021-01-01",  
  "dataFormat": "csv",  
  "blobCount": 1,  
  "byteCount": 160769,  
  "dataRowCount": 136,  
  "blobs": [  
    {  
      "blobName": "blobName.csv",  
      "byteCount": 160769,  
      "dataRowCount": 136,  
      "headerRowCount": 1,  
      "contentMD5": "md5Hash"  
    }  
  ]}
```

```
]  
}
```

Export versions

When you create a scheduled export in the Azure portal or with the API, it always runs on the exports version used at creation time. Azure keeps your previously created exports on the same version, unless you update it. Doing so prevents changes in the charges and to CSV fields if the export version is changed. As the export functionality changes over time, field names are sometimes changed and new fields are added.

If you want to use the latest data and fields available, we recommend that you create a new export in the Azure portal. To update an existing export to the latest version, update it in the Azure portal or with the latest Export API version. Updating an existing export might cause you to see minor differences in the fields and charges in files that are produced afterward.

Verify that data is collected

You can easily verify that your Cost Management data is being collected and view the exported CSV file using Azure Storage Explorer.

In the export list, select the storage account name. On the storage account page, select Open in Explorer. If you see a confirmation box, select Yes to open the file in Azure Storage Explorer.

Resource group (change)
Garda1HourBill

Status
Primary: Available, Secondary: Available

Location
North Central US, South Central US

Subscription (change)
Cost Management Demo

Subscription ID
<SubscriptionID>

Tags (change)
Click here to add tags

Services

Blobs
REST-based object storage for unstructured data
[Learn more](#)

In Storage Explorer, navigate to the container that you want to open and select the folder corresponding to the current month. A list of CSV files is shown. Select one and then select Open.

Name	Access Tier	Access Tier Last Modified	Last Modified	Blob Type	Content
CostMgmtDemo_2772a009-6687-4880-8b64-82d8c842ab69.csv			9/6/2018, 1:01:27 AM	Block Blob	appli
CostMgmtDemo_480d95a9-4b0c-44ff-ab48-4011284ff56.csv			9/8/2018, 1:02:27 AM	Block Blob	appli
CostMgmtDemo_4ace46dd-8038-442b-870a-5376a7b8d62a.csv			9/10/2018, 1:02:33 AM	Block Blob	appli
CostMgmtDemo_4c7e1c51-64e3-4729-8359-5ca1ea99e4bf.csv			9/7/2018, 1:02:36 AM	Block Blob	appli
CostMgmtDemo_bacf1aa1-fdf8-4403-9635-daa749d7af8.csv			9/9/2018, 1:02:35 AM	Block Blob	appli
CostMgmtDemo_ec45fa87-7d7e-44e8-9f8f-5d40971e3a35.csv			9/5/2018, 1:02:40 AM	Block Blob	appli

The file opens with the program or application set to open CSV file extensions. Here's an example in Excel.

A	B	C	D	E	F	G
1	DepartmentName	AccountName	AccountOwnerId	SubscriptionGuid	SubscriptionName	ResourceGroupName
2	Ama	AAAA	maeptest3@hotmail.com	1caaa5a3-2b66-43	Cost Management	Garda1HourBilling
3	Ama	AAAA	maeptest3@hotmail.com	1caaa5a3-2b66-43	Cost Management	MAR-CCM
4	Ama	AAAA	maeptest3@hotmail.com	1caaa5a3-2b66-43	Cost Management	MAR-CCM

Download an exported CSV data file

To download the CSV file, browse to the file in Microsoft Azure Storage Explorer and download it.

View export run history

You can view the run history of your scheduled export by selecting an individual export in the exports list page. The exports list page also provides you with quick access to view the run time of your previous exports and the next time and export will run. Here's an example showing the run history.

Name	Schedule status	Last run	Next run	Frequency	Storage account
daily1	Active	8/5/2020, 2:03 AM PDT	8/6/2020, 1:50 AM PDT	Daily	acmtestdiag
myexport	Active	8/5/2020, 2:03 AM PDT	8/6/2020, 2:02 AM PDT	Daily	azurecostdata
coeslalomtest	Active	8/5/2020, 2:02 AM PDT	8/6/2020, 2:01 AM PDT	Daily	usagedetailsacmdemo
aopscoetest1	Active	8/5/2020, 2:03 AM PDT	8/6/2020, 2:02 AM PDT	Daily	cloudynacmpipeline
Daily	Active	5/20/2020, 1:24 PM PDT	---	Daily	treybilling
Daily2	Active	5/20/2020, 1:24 PM PDT	---	Daily	treybilling
DailyMTD	Inactive	5/20/2020, 1:24 PM PDT	---	Daily	westusacmexport1

Select an export to view the run history.

coeslalomtest

Exports

Run now | Disable | Delete | Edit | Refresh

Scope	:	Contoso (Demo) (BillingAccount)	Storage account	:	usagedetailsacmdemo
Metric	:	Actual cost	Storage account subscrip...	:	StorageAccountSubscriptionID
Frequency	:	Daily	Storage container	:	consumptionusage
Export start date	:	9/5/2019, 11:59 AM PDT	Storage directory	:	daily
Schedule status	:	Active			

Run history

Execution time	Execution status
Aug 05, 2020, 02:02 AM	✓ Succeeded
Aug 04, 2020, 02:04 AM	✓ Succeeded
Aug 03, 2020, 02:02 AM	✓ Succeeded
Aug 02, 2020, 02:03 AM	✓ Succeeded
Aug 01, 2020, 02:02 AM	! Failed ⓘ
Jul 31, 2020, 02:04 AM	✓ Succeeded
Jul 30, 2020, 02:03 AM	✓ Succeeded
Jul 29, 2020, 02:02 AM	✓ Succeeded
Jul 28, 2020, 02:04 AM	✓ Succeeded
Jul 27, 2020, 02:02 AM	✓ Succeeded

Export runs twice a day for the first five days of the month

There are two runs per day for the first five days of each month after you create a daily export. One run executes and creates a file with the current month's cost data. It's the run that's available for you to see in the run history. A second run also executes to create a file with all the costs from the prior month. The second run isn't currently visible in the run history. Azure executes the second run to ensure that your latest file for the past month contains all charges exactly as seen on your invoice. It runs because there are cases where latent usage and charges are included in the invoice up to 72 hours after the calendar month is closed. To learn more about Cost Management usage data updates, see [Cost and usage data updates and retention](#).

ⓘ Note

Daily export created between 1st to 5th of the current month would not generate data for the previous month as the export schedule starts from the date of creation.

Access exported data from other systems

One of the purposes of exporting your Cost Management data is to access the data from external systems. You might use a dashboard system or other financial system. Such systems vary widely so showing an example wouldn't be practical. However, you can get started with accessing your data from your applications at [Introduction to Azure Storage](#).

Exports FAQ

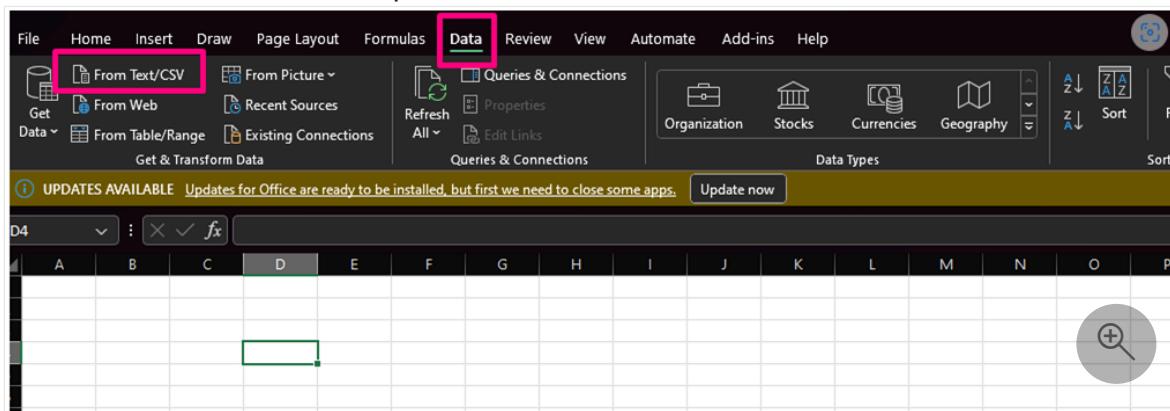
Here are some frequently asked questions and answers about exports.

Why do I see garbled characters when I open exported cost files with Microsoft Excel?

If you see garbled characters in Excel and you use an Asian-based language, such as Japanese or Chinese, you can resolve this issue with the following steps:

For new versions of Excel:

1. Open Excel.
2. Select the **Data** tab at the top.
3. Select the **From Text/CSV** option.



4. Select the CSV file that you want to import.
5. In the next box, set **File origin** to **65001: Unicode (UTF-8)**.

The dialog box shows the 'File Origin' dropdown set to '65001: Unicode (UTF-8)'. The table below displays billing profile data with columns for Column1 through Column7.

Column1	Column2	Column3	Column4	Column5	Column6	Column7
SEP=						
BILLING PROFILE NAME	BILLING PROFILE ID	STATUS	MARKUP STATUS	START DATE	EXPIRATION DATE	
National Police Agency	49437351	転送済み	Disabled	2015-05-01T00:00:00	2020-04-30T00:00:00	
H.U.グループホールディングス株式会社	64624339	Terminated	Disabled	2016-12-02T00:00:00	2017-12-31T00:00:00	
三井不動産株式会社	53323552	Terminated	Disabled	2017-01-30T00:00:00	2020-03-31T00:00:00	
P H C 株式会社	77662149	転送済み	Published	2017-03-22T00:00:00	2020-04-30T00:00:00	
株式会社テイクアンドギヴ・ニーズ	55108833	転送済み	Published	2017-04-01T00:00:00	2020-03-31T00:00:00	

6. Select Load.

For older versions of MS Excel:

1. Open Excel.
2. Select the **Data** tab at the top.
3. Select the **From Text** option and then select the CSV file that you want to import.
4. Excel shows the Text Import Wizard.
5. In the wizard, select the **Delimited** option.
6. In the **File origin** field, select **65001 : Unicode (UTF-8)**.
7. Select **Next**.
8. Next, select the **Comma** option and then select **Finish**.
9. In the dialog window that appears, select **OK**.

Why does the aggregated cost from the exported file differ from the cost displayed in Cost Analysis?

You might notice discrepancies between the aggregated cost from an exported file and the cost displayed in Cost Analysis. These differences can occur if the tool you use to read and aggregate the total cost truncates decimal values. This issue is common in tools like Power BI and Microsoft Excel.

Using Power BI

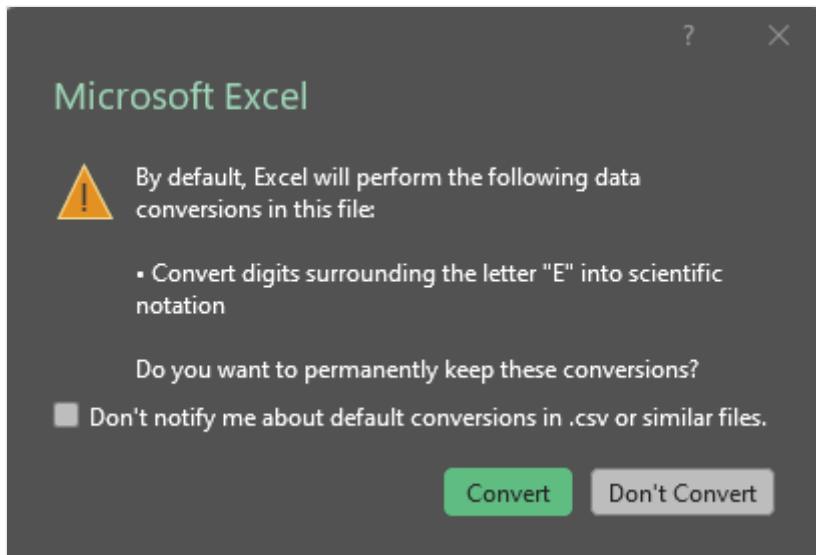
Check if decimal places are being dropped when cost values are converted into integers. Losing decimal values can result in a loss of precision and misrepresentation of the aggregated cost.

To manually transform a column to a decimal number in Power BI, follow these steps:

1. Go to the **Table** view.
2. Select **Transform data**.
3. Right-click the required column.
4. Change the type to **Decimal Number**.

Using Microsoft Excel

When you open a .csv or .txt file, Excel might display a warning message if it detects that an automatic data conversion is about to occur. Select the **Convert** option when prompted to ensure numbers are stored as numbers and not as text. It ensures the correct aggregated total. For more information, see [Control data conversions in Excel for Windows and Mac](#).



If the correct conversion isn't used, you get a green triangle with a `Number Stored as Text` error. This error might result in incorrect aggregation of charges, leading to discrepancies with cost analysis.

1 GB/Month	0.000168	0.045	7.56E-06	AC
1 GB	0.000137	6.86E-07	AC	
		Number Stored as Text	0.00018	AC
		Convert to Number	37.08	AC
		Help on this Error	8.64E-06	ac
		Ignore Error	1.72E-06	ac
		Edit in Formula Bar	0.000026	AC
		Error Checking Options...	5.26E-06	ac
			1.46E-08	AC

Next steps

In this tutorial, you learned how to:

- ✓ Create a daily export
- ✓ Verify that data is collected

Advance to the next tutorial to optimize and improve efficiency by identifying idle and underutilized resources.

[Review and act on optimization recommendations](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Get help at Microsoft Q&A](#)

Tutorial: Optimize costs from recommendations

Article • 03/21/2024

Cost Management works with Azure Advisor to provide cost optimization recommendations. Azure Advisor helps you optimize and improve efficiency by identifying idle and underutilized resources. This tutorial walks you through an example where you identify underutilized Azure resources and then you take action to reduce costs.

Watch the video [Optimizing cloud investments in Cost Management](#) to learn more about using Advisor to optimize your costs. To watch other videos, visit the [Cost Management YouTube channel](#).

<https://www.youtube-nocookie.com/embed/cSNPoAb-TNc>

In this tutorial, you learn how to:

- ✓ View cost optimization recommendations to view potential usage inefficiencies
- ✓ Act on a recommendation to resize a virtual machine to a more cost-effective option
- ✓ Verify the action to ensure that the virtual machine was successfully resized

Prerequisites

Recommendations are available for a variety of scopes and Azure account types. To view the full list of supported account types, see [Understand Cost Management data](#). You must have at least read access to one or more of the following scopes to view cost data. For more information about scopes, see [Understand and work with scopes](#).

- Subscription
- Resource group

If you have a new subscription, you can't immediately use Cost Management features. It might take up to 48 hours before you can use all Cost Management features. Also, you must have active virtual machines with at least 14 days of activity.

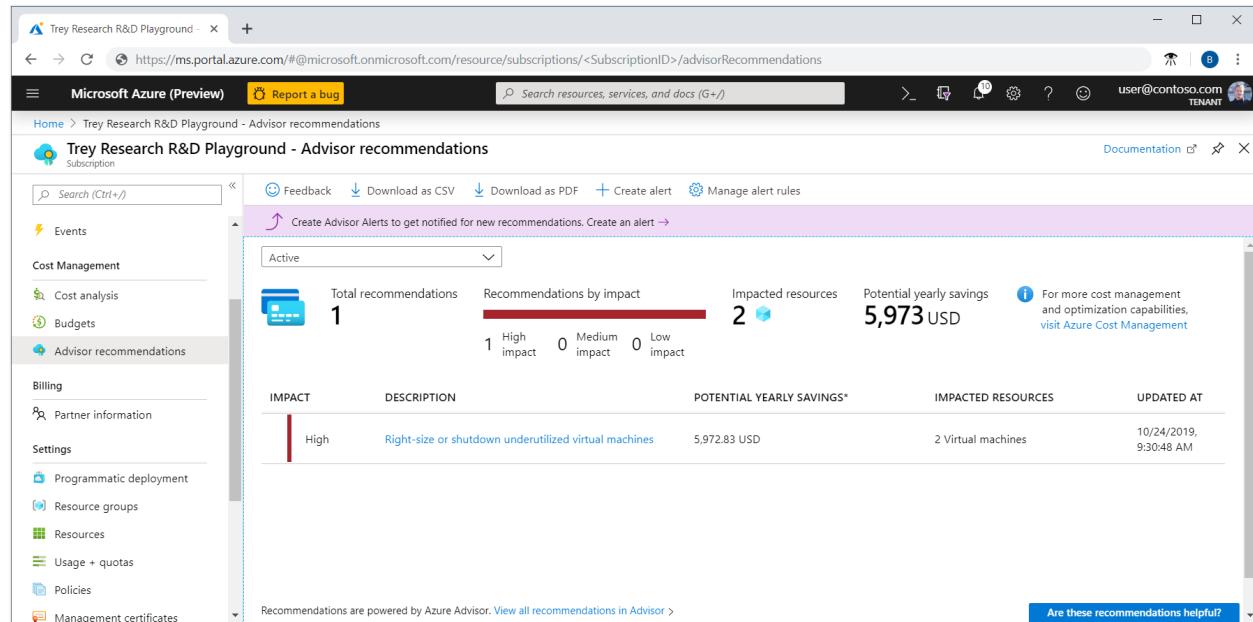
Sign in to Azure

Sign in to the Azure portal at <https://portal.azure.com>.

View cost optimization recommendations

To view cost optimization recommendations for a subscription, open the desired scope in the Azure portal and select **Advisor recommendations**.

To view recommendations for a management group, open the desired scope in the Azure portal and select **Cost analysis** in the menu. Use the **Scope** pill to switch to a different scope, such as a management group. Select **Advisor recommendations** in the menu. For more information about scopes, see [Understand and work with scopes](#).



The screenshot shows the Azure portal interface for 'Trey Research R&D Playground - Advisor recommendations'. On the left, a sidebar lists various management categories like Cost Management, Billing, and Settings. The main area displays a summary of recommendations:

- Total recommendations: 1
- Recommendations by impact:
 - High impact: 1
 - Medium impact: 0
 - Low impact: 0
- Impacted resources: 2
- Potential yearly savings: \$5,973 USD

A detailed table below provides more information for each recommendation:

IMPACT	DESCRIPTION	POTENTIAL YEARLY SAVINGS*	IMPACTED RESOURCES	UPDATED AT
High	Right-size or shutdown underutilized virtual machines	5,972.83 USD	2 Virtual machines	10/24/2019, 9:30:48 AM

At the bottom, a note states: 'Recommendations are powered by Azure Advisor. [View all recommendations in Advisor](#)'.

The list of recommendations identifies usage inefficiencies or shows purchase recommendations that can help you save additional money. The totaled **Potential yearly savings** shows the total amount that you can save if you shut down or deallocate all of your VMs that meet recommendation rules. If you don't want to shut them down, you should consider resizing them to a less expensive VM SKU.

The **Impact** category, along with the **Potential yearly savings**, are designed to help identify recommendations that have the potential to save as much as possible.

High impact recommendations include:

- Buy an Azure savings plan to save money on a variety of compute services
- Buy reserved virtual machine instances to save money over pay-as-you-go costs
- Optimize virtual machine spend by resizing or shutting down underutilized instances
- Use Standard Storage to store Managed Disks snapshots

Medium impact recommendations include:

- Reduce costs by eliminating un-provisioned ExpressRoute circuits

- Reduce costs by deleting or reconfiguring idle virtual network gateways

Act on a recommendation

Azure Advisor monitors your virtual machine usage for seven days and then identifies underutilized virtual machines. Virtual machines whose CPU utilization is five percent or less and network usage is seven MB or less for four or more days are considered low-utilization virtual machines.

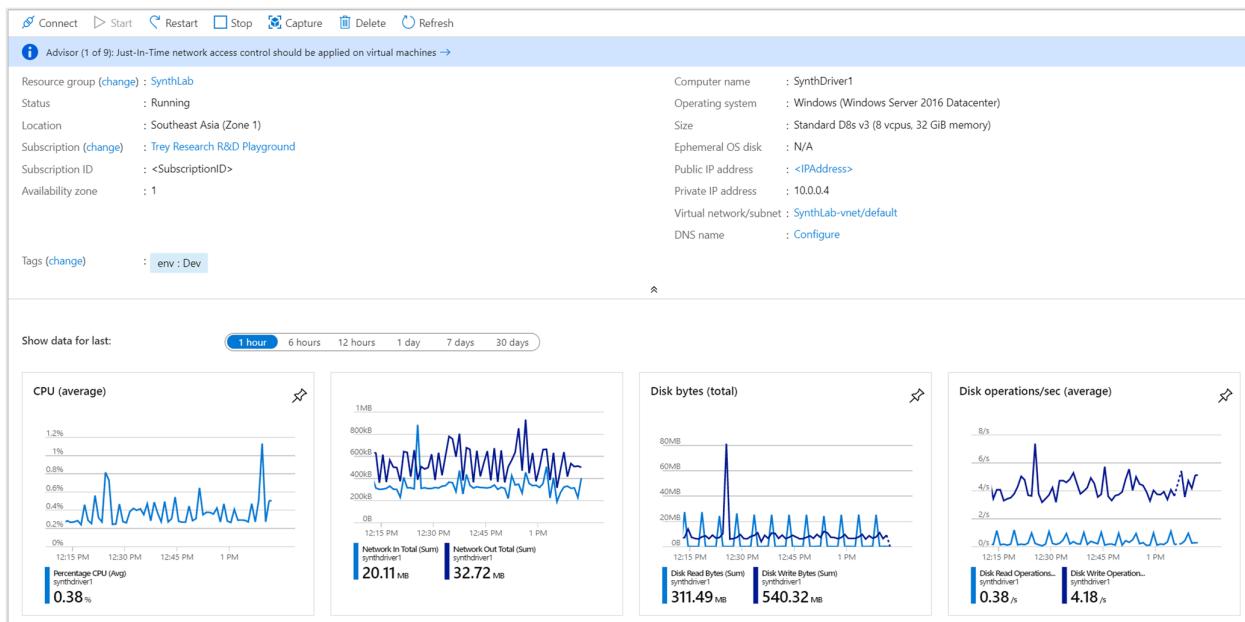
The 5% or less CPU utilization setting is the default, but you can adjust the settings. For more information about adjusting the setting, see the [Configure the average CPU utilization rule or the low usage virtual machine recommendation](#).

Although some scenarios can result in low utilization by design, you can often save money by changing the size of your virtual machines to less expensive sizes. Your actual savings might vary if you choose a resize action. Let's walk through an example of resizing a virtual machine.

In the list of recommendations, select the **Right-size or shutdown underutilized virtual machines** recommendation. In the list of virtual machine candidates, choose a virtual machine to resize and then select the virtual machine. The virtual machine's details are shown so that you can verify the utilization metrics. The **potential yearly savings** value is what you can save if you shut down or remove the VM. Resizing a VM will probably save you money, but you won't save the full amount of the potential yearly savings.

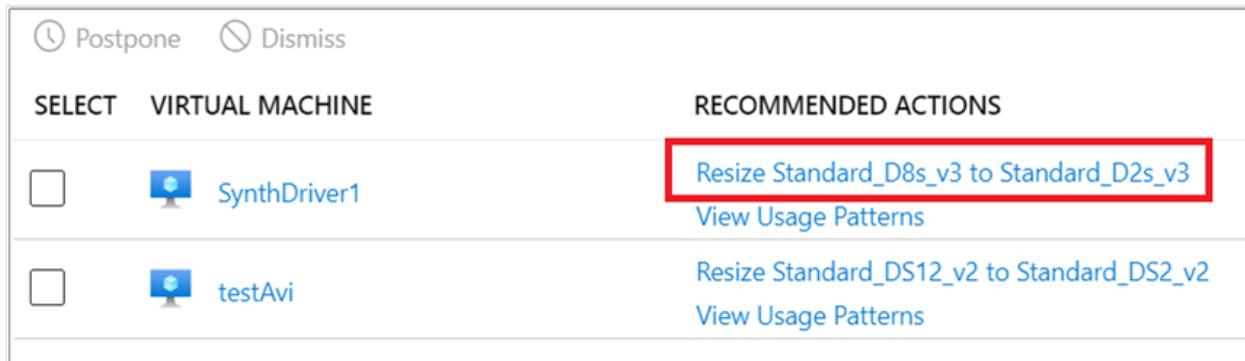
SELECT	VIRTUAL MACHINE	RECOMMENDED ACTIONS	POTENTIAL SAVINGS*	SUBSCRIPTION	RECOMMENDATION RULE	UPDATED AT	ACTION
<input type="checkbox"/>	SynthDriver1	Resize Standard_D8s_v3 to Standard_D2s_v3 View Usage Patterns	3,348.00 USD (75%)	Trey Research R&D Playground	CPU utilization < 20%	10/24/2019, 9:23:23 AM	Postpone Dismiss
<input type="checkbox"/>	testAvi	Resize Standard_DS12_v2 to Standard_DS2_v2 View Usage Patterns	2,624.83 USD (63%)	Trey Research R&D Playground	CPU utilization < 20%	10/24/2019, 9:30:48 AM	Postpone Dismiss

In the VM details, check the utilization of the virtual machine to confirm that it's a suitable resize candidate.



Note the current virtual machine's size. After you've verified that the virtual machine should be resized, close the VM details so that you see the list of virtual machines.

In the list of candidates to shut down or resize, select **Resize <FromVirtualMachineSKU> to <ToVirtualMachineSKU>**.



Next, you're presented with a list of available resize options. Choose the one that will give the best performance and cost-effectiveness for your scenario. In the following example, the option chosen resizes from **Standard_D8s_v3** to **Standard_D2s_v3**.

The screenshot shows a list of 109 VM sizes. The 'D2s_v3' row is highlighted with a red box. A 'Resize' button is visible at the bottom left.

VM Size	Offering	Family	vCPUs	RAM	Data disks	Max IOPS	Temporary storage	Premium disk support	Cost/month (estimated)
D16s_v3	Standard	General purpose	16	64	32	25600	128	Yes	\$744.00
D2_v2	Standard	General purpose	2	7	8	8x500	100	No	\$117.55
D2_v2	Promo (Exp...)	General purpose	2	7	8	8x500	100	No	\$117.55
D2_v3	Standard	General purpose	2	8	4	4x500	50	No	\$93.00
D2s_v3	Standard	General purpose	2	8	4	3200	16	Yes	\$93.00
D3_v2	Standard	General purpose	4	14	16	16x500	200	No	\$235.10

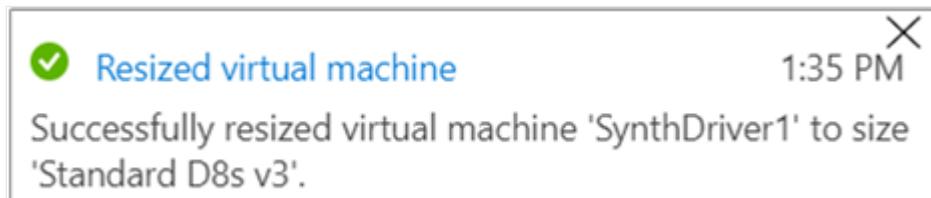
Prices presented are estimates in your local currency that include only Azure infrastructure costs and any discounts for the subscription and location. The prices don't include any applicable software costs. If you purchased Azure services through a reseller, contact your reseller for full pricing details. Final charges will appear in your local currency in cost analysis and billing views.

After you choose a suitable size, select **Resize** to start the resize action.

Resizing requires an actively running virtual machine to restart. If the virtual machine is in a production environment, we recommend that you run the resize operation after business hours. Scheduling the restart can reduce disruptions caused by momentarily unavailability.

Verify the action

When the VM resizing completes successfully, an Azure notification is shown.



Next steps

In this tutorial, you learned how to:

- ✓ View cost optimization recommendations to view potential usage inefficiencies
- ✓ Act on a recommendation to resize a virtual machine to a more cost-effective option
- ✓ Verify the action to ensure that the virtual machine was successfully resized

If you haven't already read the Cost Management best practices article, it provides high-level guidance and principles to consider to help manage costs.

[Cost Management best practices](#)

Use cost alerts to monitor usage and spending

Article • 03/21/2024

This article helps you understand and use Cost Management alerts to monitor your Azure usage and spending. Cost alerts are automatically generated based when Azure resources are consumed. Alerts show all active cost management and billing alerts together in one place. When your consumption reaches a given threshold, alerts are generated by Cost Management. There are three main types of cost alerts: budget alerts, credit alerts, and department spending quota alerts.

You can also [create a cost anomaly alert](#) to automatically get notified when an anomaly is detected.

Required permissions for alerts

The following table shows how Cost Management alerts are used by each role. The behavior below is applicable to all Azure RBAC scopes.

Expand table

Feature/Role	Owner	Contributor	Reader	Cost Management Reader	Cost Management Contributor
Alerts	Read, Update	Read, Update	Read only	Read only	Read, Update

Budget alerts

Budget alerts notify you when spending, based on usage or cost, reaches or exceeds the amount defined in the [alert condition of the budget](#). Cost Management budgets are created using the Azure portal or the [Azure Consumption API](#).

In the Azure portal, budgets are defined by cost. Using the Azure Consumption API, budgets are defined by cost or by consumption usage. Budget alerts support both cost-based and usage-based budgets. Budget alerts are generated automatically whenever the budget alert conditions are met. You can view all cost alerts in the Azure portal. Whenever an alert is generated, it's shown in cost alerts. An alert email is also sent to the people in the alert recipients list of the budget.

If you have an Enterprise Agreement, you can [Create and edit budgets](#). Customers with a Microsoft Customer Agreement should use the [Budgets REST API](#) to create budgets programmatically.

You can use the Budget API to send email alerts in a different language. For more information, see [Supported locales for budget alert emails](#).

Credit alerts

Credit alerts notify you when your Azure Prepayment (previously called monetary commitment) is consumed. Azure Prepayment is for organizations with Enterprise Agreements. Credit alerts are generated automatically at 90% and at 100% of your Azure Prepayment credit balance. Whenever an alert is generated, it's reflected in cost alerts and in the email sent to the account owners.

Department spending quota alerts

Department spending quota alerts notify you when department spending reaches a fixed threshold of the quota. Spending quotas are configured in the Azure portal. Whenever a threshold is met it generates an email to department owners and is shown in cost alerts. For example, 50% or 75% of the quota.

Supported alert features by offer categories

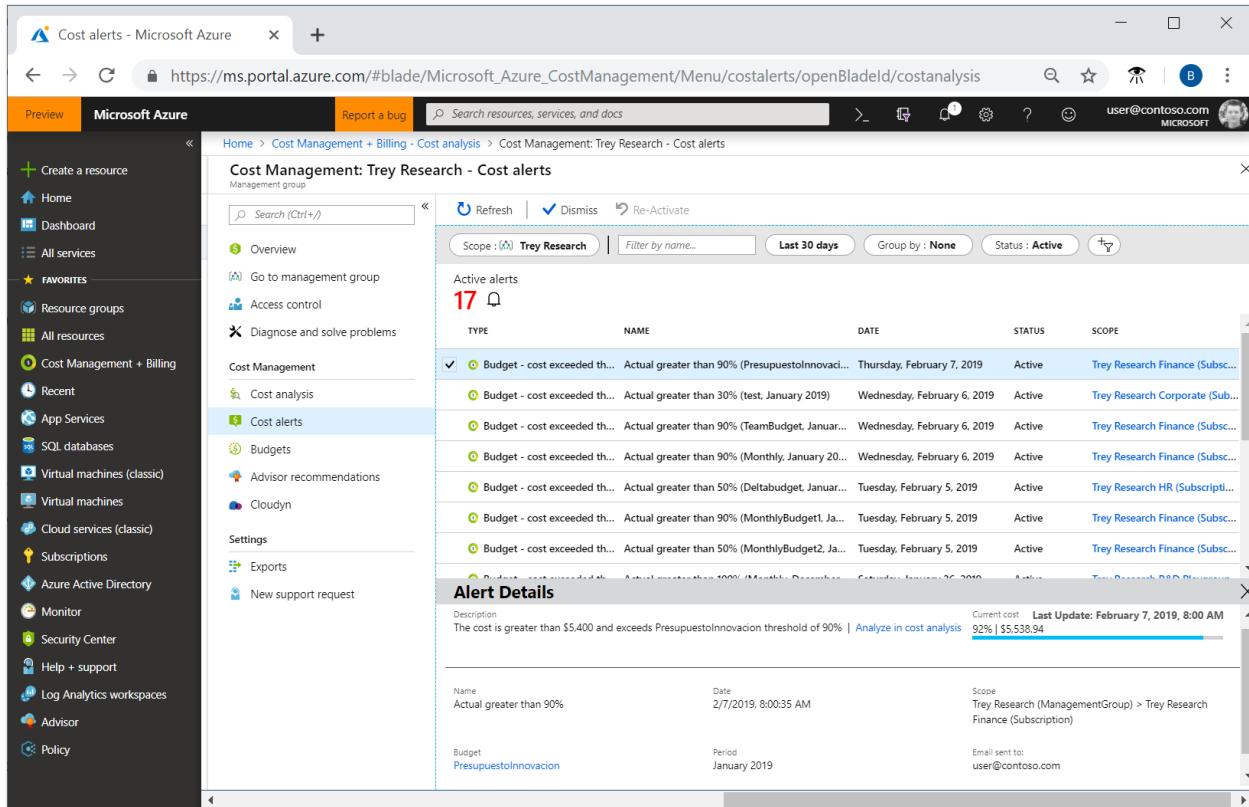
Support for alert types depends on the type of Azure account that you have (Microsoft offer). The following table shows the alert features that are supported by various Microsoft offers. You can view the full list of Microsoft offers at [Understand Cost Management data](#).

[] Expand table

Alert type	Enterprise Agreement	Microsoft Customer Agreement	Web direct/Pay-As-You-Go
Budget	✓	✓	✓
Credit	✓	✗	✗
Department spending quota	✓	✗	✗

View cost alerts

To view cost alerts, open the desired scope in the Azure portal and select **Budgets** in the menu. Use the **Scope** pill to switch to a different scope. Select **Cost alerts** in the menu. For more information about scopes, see [Understand and work with scopes](#).



The screenshot shows the Microsoft Azure portal interface. The left sidebar is the navigation menu with various service icons. The main content area is titled "Cost Management: Trey Research - Cost alerts". A sub-menu on the left under "Cost Management" includes "Cost analysis", "Cost alerts" (which is selected), "Budgets", "Advisor recommendations", and "Cloudyn". The main pane displays a table of "Active alerts" with 17 entries. The columns are "TYPE", "NAME", "DATE", "STATUS", and "SCOPE". Each row shows a green circular icon followed by a brief description of the alert type, the name of the budget it applies to, the date it was generated, its status, and the scope it applies to. Below the table, there is a section titled "Alert Details" with fields for "Name" (Actual greater than 90%), "Date" (2/7/2019, 8:00:35 AM), "Scope" (Trey Research (ManagementGroup) > Trey Research Finance (Subscription)), "Budget" (PresupuestolInnovacion), "Period" (January 2019), and "Email sent to" (user@contoso.com). There is also a link to "Analyze in cost analysis".

The total number of active and dismissed alerts appears on the cost alerts page.

All alerts show the alert type. A budget alert shows the reason why it was generated and the name of the budget it applies to. Each alert shows the date it was generated, its status, and the scope (subscription or management group) that the alert applies to.

Possible status includes **active** and **dismissed**. Active status indicates that the alert is still relevant. Dismissed status indicates that someone has marked the alert to set it as no longer relevant.

Select an alert from the list to view its details. Alert details show more information about the alert. Budget alerts include a link to the budget. If a recommendation is available for a budget alert, then a link to the recommendation is also shown. Budget, credit, and department spending quota alerts have a link to analyze in cost analysis where you can explore costs for the alert's scope. The following example shows spending for a department with alert details.

The screenshot shows the Azure portal's alert management interface. At the top, there are buttons for Refresh, Dismiss, and Re-Activate. The scope is set to Contoso (Demo) (8608480). Below this, a summary shows 3 Active alerts and 1 Dismissed alert. A table lists four alerts, with the first three being active and the fourth being dismissed. The dismissed alert is for 'Azure credit (system notification)' with the message 'You have used over 100% of your azure credits'. The alert details pane is open, showing the alert name, date (1/15/2019, 5:02:51 PM), and scope (Contoso (Demo) (8608480) (BillingAccount) > ACE (Department)). It also shows current cost (2193%) and last update (February 4, 2019, 5:00 PM). A recommendation to 'Analyze in cost analysis' is present.

When you view the details of a dismissed alert, you can reactivate it if manual action is needed. The following image shows an example.

This screenshot shows the Azure portal's alert management interface for the scope 'Trey Research Finance'. It displays 4 Active alerts and 1 Dismissed alert. The dismissed alert is for 'Budget - cost exceeded threshold' with the message 'Actual greater than 90% (Monthly, January 2019)'. The 'Re-Activate' button for this alert is highlighted with a red box. The alert details pane is open, showing the alert name, date (1/15/2019, 5:02:51 PM), and scope (Contoso (Demo) (8608480) (BillingAccount) > ACE (Department)). It also shows current cost (2193%) and last update (February 4, 2019, 5:00 PM). A recommendation to 'Analyze in cost analysis' is present.

See also

- If you haven't already created a budget or set alert conditions for a budget, complete the [Create and manage budgets](#) tutorial.

Keep your Azure landing zone up to date

Article • 04/17/2024

An Azure landing zone is a set of pre-defined Azure resources and configurations that provide a foundation for a cloud-based application or workload. It's important to ensure that your deployed landing zone environment is up to date so that you can maintain improved security, avoid platform configuration drift, and stay optimized for new feature releases.

<https://www.youtube-nocookie.com/embed/VvZDftlF20w> ↗

Why update your Azure landing zones?

Here are a few reasons to keep your landing zone up to date:

- **Maintain improved security.** Cybersecurity threats are constantly evolving. It's important to ensure that your landing zone reflects the latest best practices for protecting your data and systems. Keeping your landing zone up to date helps you mitigate the risk of a security breach and helps you keep your data properly secured.
- **Avoid platform configuration drift.** As landing zones continue to evolve, drift relative to your deployed environment is introduced. Examples of drift include:
 - Replacement of landing zone policies by built-in Azure policies or by newer versions of landing zone policies.
 - Improvements to network features.
 - New features.

The longer drift is left unattended, the more technical debt it incurs. This debt requires remediation. So that you can avoid spending increased time on remediation activities, we encourage you to regularly review the latest [changes to landing zones](#) ↗ .

- **Optimize for Azure improvements.** As new Azure features and services are released, landing zones might be modified to include them. Likewise, as older Azure features are deprecated, changes might also be made to landing zones.
- **Get support.** A landing zone, as a deployable reference and implementation, is an open-source project, so support is limited to community engagement. Keeping

your landing zone aligned to the current implementation makes community support more likely.

Neglecting to keep your landing zones up to date could affect your security posture and the benefits that you get from the landing zones. To protect your investment in Azure, regularly review and update your landing zones as needed. See the **Next steps** section for guidance on how to do that.

Keep policies and policy initiatives up to date

Over time, Azure landing zone custom policies and policy initiatives might be updated to newer versions or even superseded by new Azure built-in policies. If so, they should be included in your platform landing zone update cycle.

- Migrate landing zone custom policies to Azure built-in policies
- Update Azure landing zone custom policies
- Advanced policy management using [Enterprise Policy as Code \(EPAC\)](#)

Use infrastructure as code (IaC) to keep ALZ updated

Maintain your ALZ environment with IaC to consistently stay updated with ALZ. To learn more about the benefits and details, see [Use infrastructure as code to update Azure landing zones](#).

Next steps

- [Latest updates to landing zones ↗](#)
-

Feedback

Was this page helpful?



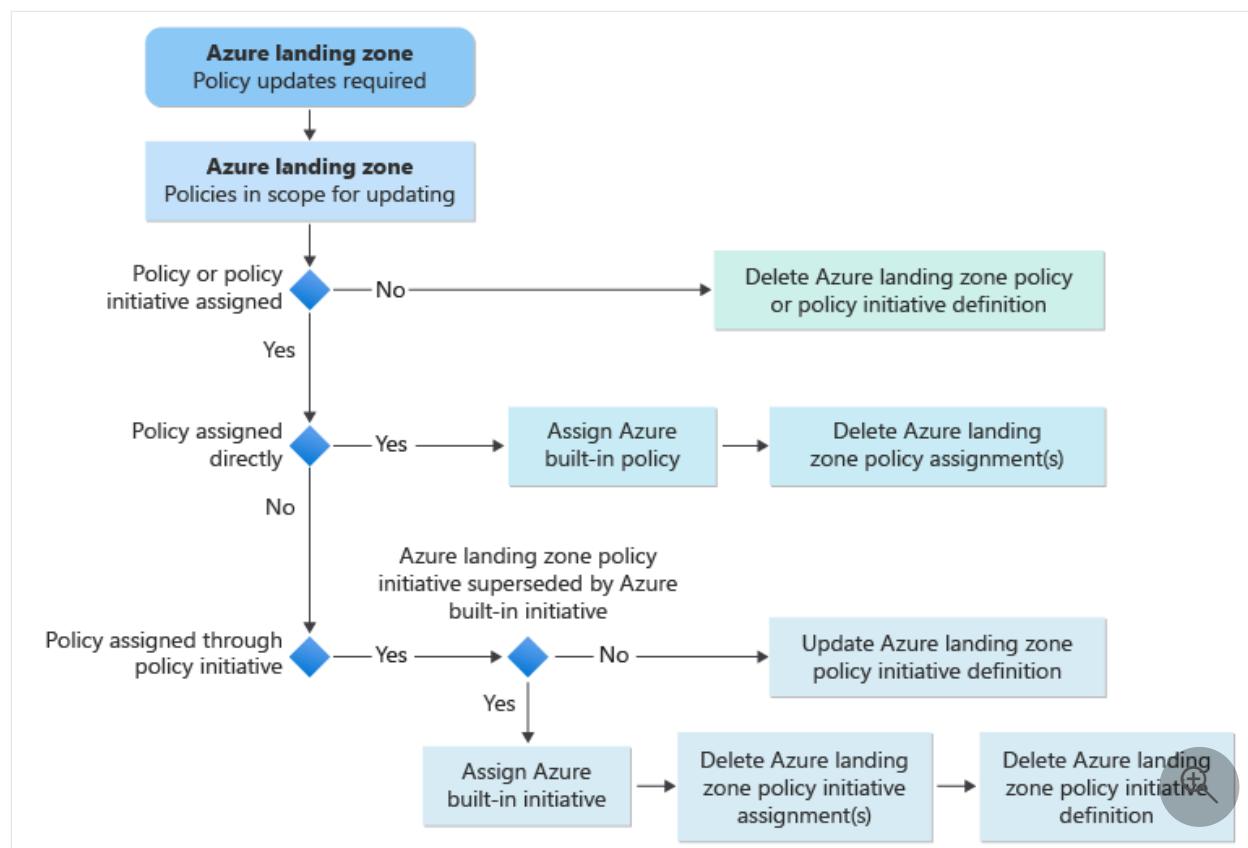
Azure landing zone governance guide: Migrate Azure landing zone policies to Azure built-in policies

Article • 04/17/2024

Over time, Azure landing zone custom policies and policy initiatives might be deprecated or superseded by Azure built-in policies. If so, they should be removed or migrated. This article describes how to migrate Azure landing zone custom policies and policy initiatives to Azure built-in policies.

The guidance in this document describes the manual, high-level steps for your policies migration. It also provides references on how to process implementations managed through the [Azure landing zone Terraform module](#) or [ALZ-Bicep](#).

The following infographic shows the update process flow.



Manual update steps for Azure landing zone environments

This section describes the generic, high-level steps to migrate Azure landing zone custom policies and initiatives to Azure built-in policies.

Detect updates for Azure landing zone policies

You can detect that one or more Azure landing zone policies are superseded by built-in Azure policies with the following options:

- Periodically review [Azure Enterprise Scale What's new wiki](#) and note any policies indicated as being superseded. See [an example of a superseded policy here](#).
- Use the [Azure Governance Visualizer](#) script and note any policies marked as obsolete.

Migration steps for Azure landing zone policies

You can migrate Azure landing zone environments with the following steps:

1. Determine if the Azure landing zone policies in scope for migration are currently assigned at any scope in your Azure estate. If you're using the [Azure Governance Visualizer](#), you can determine policy scope by checking the [TenantSummary](#).
2. Check if the Azure landing zone policies being migrated are part of a landing zone custom policy initiative that should be updated.
3. See if Azure landing zone custom policy initiatives in scope for migration are currently assigned at any scope in your Azure estate.

Depending on the results of your investigation, take the following actions.

Policies not assigned and not part of Azure landing zone custom policy initiative

If the policy being migrated isn't assigned in your Azure estate, and isn't part of an existing Azure landing zone custom policy initiative, you:

- Delete the Azure landing zone policy definition from the Azure landing zone intermediate root management group (for example, `Contoso`).

If an Azure landing zone custom policy initiative is fully superseded by a built-in policy initiative and isn't assigned in your Azure estate, you:

- Delete the Azure landing zone custom policy initiative from the Azure landing zone intermediate root management group (for example, `Contoso`).

Policies assigned and not part of Azure landing zone custom policy initiative

If the policy to be migrated is assigned to any scope in your Azure estate, and isn't part of an existing Azure landing zone custom policy initiative, do these steps:

1. Create new policy assignments at the same scopes using the Azure built-in policies with matching settings as per the assignment of the previous Azure landing zone custom policy definition.
2. Delete existing Azure landing zone policy assignment at all scopes, where assigned.
3. Delete the Azure landing zone policy definition from the Azure landing zone intermediate root management group (for example `Contoso`).

For detailed guidance on how to do the previous steps, see [Migrate single Azure landing zone custom policy](#).

Policies assigned through Azure landing zone custom policy initiative

If the policy to be migrated is part of an Azure landing zone custom policy initiative and is assigned through it at any scope in your Azure estate, follow these steps:

1. Update the Azure landing zone custom policy initiative definition with the appropriate policy references. You can find the [updated initiatives here](#) with a generic `contoso` scope for custom policies.
2. When you update the policy references, remember to change the `contoso` scope for policy definition IDs to your management group hierarchy pseudo root name. Also, update the metadata information on the Azure landing zone custom policy initiative.

For detailed guidance on how to do the previous steps, see [How to update child definitions in Azure landing zone custom initiatives](#).

If an Azure landing zone custom policy initiative is fully superseded by a built-in policy initiative, and assigned at any scope in your Azure estate, follow these steps:

1. Create new policy initiative assignments at the same scopes. Use the Azure built-in policy initiative with matching settings per the assignment of the previous Azure landing zone custom policy initiative.
2. Delete existing Azure landing zone policy initiative assignment at all scopes, where assigned.
3. Delete the Azure landing zone custom policy initiative from the Azure landing zone intermediate root management group (for example, `Contoso`).

Update steps for Azure landing zone Terraform module deployments

If you're using the [Azure landing zone Terraform module](#) to manage your Azure landing zone deployment, this section references resources on how to migrate Azure landing zone custom policies and initiatives to Azure built-in policies.

Detect updates for Terraform module changes

Use the methods described in [Detect updates for Azure landing zone policies](#) to determine whether policies have changed in the Terraform module. You'll also see changes to policies in the [Azure landing zone Terraform releases page](#). See an example [here](#).

Migration steps for Azure landing zone Terraform module

The Azure landing zone Terraform module provides update guidance when deploying breaking changes. Follow the upgrade guidance available for your specific version [here](#) at the bottom of the page.

Update steps for ALZ-Bicep deployments

If you're using the [ALZ-Bicep](#) to manage your Azure landing zone deployment, this section references resources on how to migrate Azure landing zone custom policies and initiatives to Azure built-in policies.

Detect updates for ALZ-Bicep policy changes

Use the methods described in [Detect updates for Azure landing zone policies](#) to determine whether policies have changed in ALZ-Bicep. You'll also see changes to policies in [ALZ-Bicep releases](#).

Migration steps for ALZ-Bicep policies

ALZ-Bicep provides generic guidance for migrating policies from Azure landing zone custom policies to Azure built-in policies. For more information, see [How to migrate Azure landing zone custom policies to Azure built-in policies](#).

Next steps

Regardless of whether you use Azure portal, Bicep, or Terraform to manage your Azure landing zone infrastructure, policies change over time. You'll need to manage them. Use the flow described in this article as a starting point to develop processes around policy management for your specific Azure landing zone implementation.

Feedback

Was this page helpful?

 Yes

 No

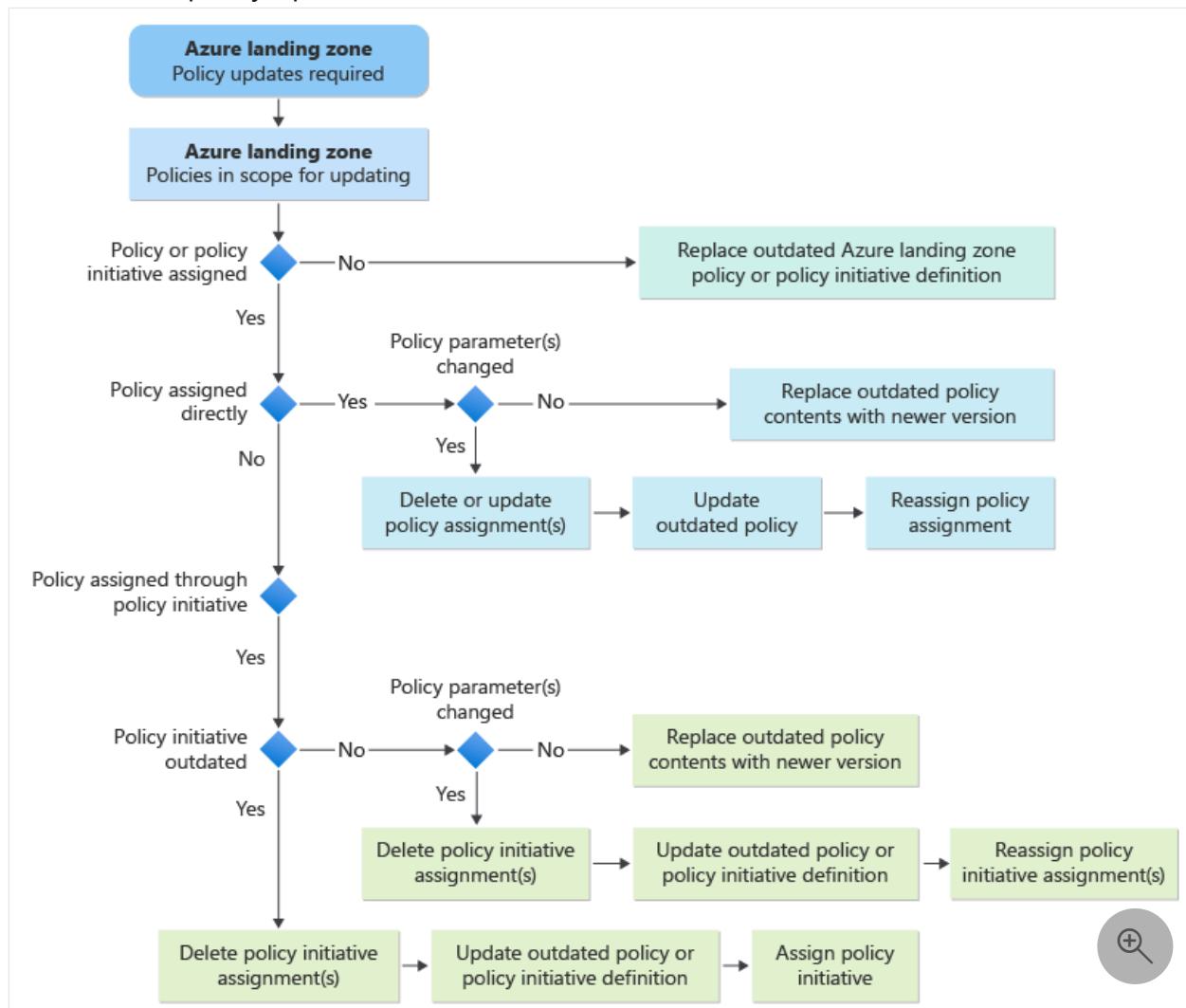
Update Azure landing zone custom policies

Article • 04/17/2024

Over time, Azure landing zone custom policies and policy initiatives update to newer versions that you can incorporate into your Azure environment. This article describes how to update your Azure landing zone custom policies and policy initiatives when newer versions release.

The article describes high-level manual update steps, and provides references on handling updates for [Terraform](#) and [Bicep](#) modular implementations. To migrate Azure landing zone custom policies to Azure built-in policies with Bicep, see [Migrate Azure landing zone policies to Azure built-in policies](#).

The following infographic provides a decision tree and process flow for Azure landing zone custom policy updates:



⊗ Caution

When you delete existing policy assignments, your environment isn't protected during the time it takes to reassign policies. After you assign updated policies, review your policy compliance section for any unhealthy resources and remediate them.

Update steps for Azure landing zone environments

This section describes the general high-level steps to update your Azure landing zone custom policies and initiatives to newer versions.

Detect updates

Use the following authoritative options to determine that one or more of your Azure landing zone custom policies are outdated:

- Periodically review [What's New](#), and note that one or more policies are updated, such as [this example](#).
- Use the [Azure Governance Visualizer](#) and note that one or more policies are marked as outdated.

Apply updates

To determine whether to apply updated custom policies to your Azure landing zone deployment:

1. Determine whether your Azure estate currently assigns any outdated custom policies at any scope. If you use the [Azure Governance Visualizer](#), you can see your currently assigned policies by checking the [TenantSummary](#).
2. Determine whether any of the outdated custom policies are part of an Azure landing zone custom policy initiative.
3. Determine whether your Azure estate currently assigns any of the outdated custom policy initiatives at any scope.

Depending on the result of the above investigations, take the following actions:

Policies not assigned

- If the outdated policy isn't assigned in your Azure estate, and isn't part of an existing custom policy initiative, replace the outdated policy definition with the

updated policy definition at the Azure landing zone intermediate root management group, such as `Contoso`.

- If a custom policy initiative is updated, but isn't assigned in your Azure estate, replace the outdated custom policy initiative with the updated custom policy initiative at the Azure landing zone intermediate root management group, for example `Contoso`.

Policies with unchanged parameters and not part of a custom policy initiative

If the outdated Azure landing zone policy is assigned to any scope in your Azure estate, isn't part of an existing Azure landing zone custom policy initiative, and the parameter names and number haven't changed:

- Replace the existing custom policy definition contents with the updated custom policy definition contents at the Azure landing zone intermediate root management group, for example `Contoso`. For detailed guidance, see the [Azure landing zones user guide](#).

Policies with changed parameters and not part of a custom policy initiative

If the outdated Azure landing zone policy is assigned to any scope in your Azure estate, isn't part of an existing Azure landing zone custom policy initiative, and the parameter names and number have changed:

1. Capture all outdated policy assignments, where they're assigned, and their parameter values.
2. Take one of these actions:
 - If the policy assignment includes more than one policy definition, update the policy assignment by removing the outdated policy at all scopes where assigned.
 - If the policy assignment contains only the outdated policy, delete the existing policy assignment at all scopes where assigned.
3. Delete the outdated policy from the Azure landing zone intermediate root management group, for example `Contoso`.
4. Import the updated policy to the Azure landing zone intermediate root management group.

5. Update the existing policy assignments or create new policy assignments by including the updated policy at the prerecorded scopes.
6. After you reassign the updated custom policy, review the policy compliance section to validate that resources are in a healthy state.

For detailed guidance, see the [Azure landing zones user guide](#).

Policies with unchanged parameters assigned through a custom policy initiative

If the outdated Azure landing zone policy is part of an existing Azure landing zone custom policy initiative, is assigned to any scope in your Azure estate, and has unchanged parameter names and numbers:

- Replace the existing custom policy definition contents with the updated custom policy definition contents. No further changes need to be made to the custom policy initiative or assignments, because the parameter number and names are unchanged. For detailed guidance, see the [Azure landing zones user guide](#).

Policies with changed parameters assigned through a custom policy initiative

If the outdated policy is part of an existing custom policy initiative, is assigned to any scope in your Azure estate, and has changed parameter names and numbers:

1. Capture all policy assignments, where they're assigned, and their parameter values for the custom policy initiative.
2. Delete the existing policy assignments at all scopes where assigned.
3. Delete the outdated policy from the custom policy initiative.

You can't delete *initiative parameter(s)* from the custom policy initiative. Consider reusing these parameters.

4. Delete the outdated policy from the Azure landing zone intermediate root management group, for example `Contoso`.
5. Import the updated policy to the Azure landing zone intermediate root management group.
6. Add the updated policy to the custom policy initiative.
 - If applicable, reuse the previous initiative parameters.

- If applicable, add other initiative parameters by following existing naming patterns that the custom policy initiative defines.

7. Reassign the updated custom policy initiative.

8. After you reassign the updated custom policy initiative, review the policy compliance section to validate that resources are in a healthy state.

For detailed guidance, see the [Azure landing zones user guide](#).

Updated assigned custom policy initiative

If an Azure landing zone custom policy initiative is completely updated, and is assigned at any scope in your Azure estate:

1. Capture all policy assignments, where they're assigned, and their parameter values for the Azure landing zone custom policy initiative.
2. Delete the existing policy assignments at all scopes where assigned.
3. Delete the outdated custom policy initiative from the intermediate root management group, for example `Contoso`. Before deleting, record all custom policy definition names and IDs, assuming all custom policy definitions are up-to-date.
4. Import the updated custom policy initiative definition with the appropriate policy references.

You can get updated initiatives at [policySetDefinitions](#), with a generic `contoso` scope for custom policies. Remember to change the `contoso` scope to your management group hierarchy pseudo root name for each policy definition ID.

5. Reassign the updated custom policy initiative.

6. After you reassign the updated custom policy initiative, review the policy compliance section to validate that resources are in a healthy state.

For detailed guidance, see the [Azure landing zones user guide](#).

Update steps for Terraform module deployments

If you use the [Azure landing zones Terraform module](#) to manage your Azure landing zone deployment, this section provides resources for updating Azure landing zone

custom policies and initiatives.

Detect updates with Terraform

Use the methods in [Detect updates](#) to determine whether policies have changed. In the Terraform module, you can also see changes to policies on the [releases](#) page. For an example, see [policy updates for v2.3.0](#).

Update with Terraform

The Azure landing zone Terraform module provides update guidance for deploying breaking changes. Follow the upgrade guidance for your version at [upgrade guides](#).

Update steps for Bicep module deployments

If you use the [ALZ-Bicep modules](#) to manage your Azure landing zone deployment, this section provides resources for updating Azure landing zone custom policies and initiatives.

Detect updates with Bicep

Use the methods in [Detect updates](#) to determine whether policies have changed. You can also see changes to ALZ-Bicep policies in [ALZ-Bicep releases](#).

Update with Bicep

ALZ-Bicep provides generic guidance for updating Azure landing zone custom policies to newer policies. For more information, see [How to migrate Azure landing zone custom policies to Azure built-in policies](#).

Next steps

Regardless of whether you use the Azure portal, Bicep, or Terraform to manage your Azure landing zone infrastructure, you need to manage policy changes over time. Use the flow in this article as a starting point to develop processes around policy management for your Azure landing zone implementation.

Feedback

Was this page helpful?

 Yes

 No

Platform automation and DevOps

Article • 02/28/2023

Platform automation and DevOps evaluate opportunities to modernize your approach to environmental deployment with infrastructure as code options.

Design area review

Involved roles or functions: Platform automation and DevOps might require support from one of the following functions or roles to make decisions: [cloud platform](#) and [cloud center of excellence](#).

Scope: The goal of platform automation and DevOps is to align your desired DevOps principles and practices to Azure Landing Zone lifecycle management. This goal includes provisioning, management, evolution, and operations through extreme automation and Infrastructure as Code.

Design area overview

The scale, agility, and flexibility part of cloud technologies leads to opportunities for new ways of working and modern approaches to service delivery.

Many traditional IT operating models aren't compatible with the cloud and must undergo operational transformation to deliver against enterprise migration targets. You can evaluate using DevOps processes and tools for application and central teams.

Platform automation

The ability to make changes at scale through a prescribed automated process provides direct benefits to the organization's ability to expand beyond the baseline configuration, which comes from security, governance, and management.

Platform automation is directly applicable to the outcomes associated with implementing a landing zone, and supports the concept of building repeatable, scalable environments.

- [Automation](#) focuses on tools and techniques that enable the streamlining of automation tasks for Azure Landing Zone development, deployment, provisioning, and operations using automation tools such as Azure DevOps Services or GitHub.

DevOps

These resources address platform automation for DevOps.

- [DevOps considerations](#) explores the need to have a clear and common understanding of DevOps in the organization. This resource also describes DevOps principles, practices, and capabilities that apply to workloads and to landing zones.
- [DevOps teams topologies](#) describes how teams can be organized to own the end-to-end lifecycle of the Azure Landing Zone. Learn how these teams collaborate with other teams in the organization responsible for the end-to-end lifecycle of workloads deployed to Azure.

Development strategy

These resources address development strategy.

- [Development lifecycle](#) explores key design considerations and recommendations for the creation of a landing zone by using automation. This resource discusses the repository, branch, automated builds, deployment, and rollback strategy.
- [Infrastructure as Code](#) explains the benefits of implementing Azure Landing Zones by using Infrastructure as Code. Learn about considerations around code structure, tools, and technology.
- [Environments](#) explains the purpose of multienvironments to build, test, and release code with greater speed and frequency. This approach makes deployment as straightforward as possible.
- [Test-driven development](#) addresses how to use unit testing to improve the quality of new features and improvements in the Azure Landing Zone code-base.

Security considerations

These resources address security considerations in platform automation.

- [Security considerations](#) addresses security and governance considerations for the DevOps lifecycle of Azure Landing Zones.
- [Role-based Access Control for DevOps Tools](#) explains the access control considerations to be considered when addressing Azure Landing Zones lifecycle through DevOps tools.

Next steps

Automation

Automation

Article • 05/18/2023

In software defined cloud infrastructure, teams use various tools and techniques to provision, configure, and manage the infrastructure. As your teams evolve and grow, they can transition from using portals and manual efforts to using code and automation to provision, configure, and manage infrastructure and services.

Platform automation considerations

- Implementing the Everything as Code (EaC) methodology allows your teams to unlock key benefits, create a strong development culture, and enable everyone on each team to inspect how and which resources are deployed. EaC also helps your platform teams adopt key development practices that improve their agility and efficiency. Your teams can track changes and control which ones move to production by housing code in repositories and using version control systems to manage it.
- Teams can follow the **4-eyes principle** and use *peer programming* or *peer review* to ensure code changes are never made alone. Peer programming and peer review improve code quality, let teams share responsibility for changes, and increase team knowledge about what is agreed upon and deployed. Code review is a fantastic way for team members to learn new techniques and methods for coding and automation.
- Teams should use version control systems like Git, together with [Git repositories](#), to enforce peer review. Git repositories let your teams define important branches and protect them with [branch policies](#). You can use policy to require code changes on these branches to meet certain criteria, like a minimum number of team member approvals, before they can merge into a protected branch.
- Teams should connect the EaC methodology and the change review process together with a [continuous integration and continuous delivery \(CI/CD\)](#) process. Every code change should automatically trigger a CI process that executes static code analysis, validation, and test deployments. CI ensures that developers check their code early (often referred to as **fail fast** or **shift-left testing**) for errors that can cause future issues. Depending on which [branching strategy](#) your team uses, changes to any important branch should trigger deployment to different [environments](#). Once changes are approved and merged into [main](#), the CD process deploys those changes to production. This code management system provides your team with a **single source of truth** for what is running in each environment.

- To ensure your platform is fully self-healing and provides self-service for your workload teams, your platform team must work to automate everything (often referred to as **Extreme Automation**) from provisioning, configuration, and platform management to landing zone subscription provisioning for workload teams. Extreme automation allows your platform team to focus on providing value rather than on deploying, configuring, and managing your platform. Extreme automation also creates a self-enhancing cycle that gives your team more time to build more automation.
- As your platform teams automate operational activities and reduce human intervention, they should shift their focus to important tasks that enable and accelerate workload team innovation on Azure. To achieve this, your platform team must iterate through multiple cycles of building and development as they put into place your platform's tools, scripts and capability enhancements.
- There are multiple options available to help your team get started with their Azure Landing Zone deployment. These options depend on current team capabilities and can grow as your team evolves. More specifically, for [Platform Deployment](#), one can choose between Portal, Bicep or Terraform-based experiences, depending on the respective Teams' IaC proficiency and tooling preference.
 - New and emerging platform teams that are still getting to know [Infrastructure as Code \(IaC\)](#) and are more familiar with using a portal to deploy and manage resources can use the [Azure landing zone accelerator](#) to start, which supports teams still using a **ClickOps** approach. ClickOps is the process of provisioning, configuring, and managing resources by clicking in portals, management consoles, and wizards. This accelerator allows your team to use the portal as initial deployment tool, and progressively, as platform engineering maturity grows, to further use Azure CLI, PowerShell or IaC.
 - The [AzOps solution](#) allows teams to evolve their platform automation and management practices from **ClickOps** driven to **DevOps** capable. Your team can transition from using their personal account access to using DevOps principles and practices relying only on CI/CD with AzOps and IaC. AzOps lets your team bring their own architecture, use the architecture deployed by Azure Landing Zone Portal accelerator (after the initial portal-based deployment, as AzOps integration is not part of the ALZ Portal experience), integrate with a brownfield deployment, or use custom templates (Bicep or ARM) to build and operationalize your platform.
 - Platform teams with established skills and capabilities can adopt a codified approach that follows [DevOps principles and practices](#). Your team should base themselves heavily on IaC and modern development practices, transitioning away from using Azure access on their personal accounts and toward running all operations through your CI/CD pipeline. Your team should use IaC-based

accelerators, like [ALZ-Bicep](#) or the [Azure landing zones Terraform module](#) to accelerate this transition.

- IaC-based accelerators have limited management scope. New versions provide more capabilities and increased resource management ability. If using an accelerator, your team should consider a layered approach that starts with an accelerator, then adds a layer of automation. The automation layer provides capabilities your team needs in order to fully support your workload teams with platform features like domain controller deployment for legacy applications.
- As your platform team transitions to a DevOps approach, they need to establish a process for handling emergency fixes. They can use [Privileged Identity Management \(PIM\)](#) eligible permissions to request access to perform fixes and later bring it back to code to limit configuration drift, or they can use code to implement a quick fix. Your team should always register quick fixes in their backlog so they can rework each fix at a later point and limit their technical debt. Too much technical debt leads to future deceleration since some platform code isn't fully reviewed and doesn't meet team coding guidelines and principles.
- You can use [Azure Policies](#) to add some automation to your platform. Consider using IaC to deploy and manage Azure Policies, often referred to as Policy-as-Code (PaC). These policies let you automate activities like log collection. Many PaC frameworks also implement an exemption process, so plan for your workload teams to request exemptions from policies.
- Use "Policy-driven-governance" to signal to workload teams when they're attempting to deploy resources that don't meet a security control. Consider deploying policies with the `deny` effect for these situations, which allows your workload teams to also treat Everything as Code and avoid configuration drift where code declares one thing and policy changed a setting at deployment time. Avoid using `modify` effects, such as if a workload team deploys a storage account with `supportOnlyHttpsTraffic = false` defined in their code where a `modify` policy changes that to `true` at deployment time to keep it compliant. This leads the code drift from what is deployed.

Platform automation design recommendation

- Follow an **Everything as Code** approach for full transparency and configuration control of the Azure platform, documentation, deployment, and testing process.
- Use [version control](#) to manage all your code repositories, including:
 - Infrastructure as Code
 - Policy as Code
 - Configuration as Code
 - Deployment as Code

- Documentation as Code
- Implement **the 4-eyes principle** and a process for *peer-programming* or *peer-review* to ensure that all code changes are reviewed by your team before being deployed to production.
- Adopt a [branching strategy](#) for your team and [set branch policies](#) for branches that you want to protect. With branch policies, teams must use [pull requests](#) to make merge changes.
- Use [continuous integration and continuous delivery \(CI/CD\)](#) to automate code testing and deployment to different environments.
- Work to **automate everything**, such as the provisioning, configuration, and management of your platform and the provisioning of landing zone subscriptions for your workload teams.
- Use one of the available accelerators that matches your team's capabilities to get started with deploying Azure Landing Zones.
- Plan to use a layered deployment approach to add capabilities that aren't covered by an accelerator but are needed to fully support your workload teams.
- Establish a process for using code to implement quick fixes. Always register quick fixes in your team's backlog so each fix can be reworked at a later point and you can limit technical debt.
- Use [Infrastructure as Code](#) to deploy and manage [Azure Policies](#) (often referred to as Policy-as-Code)
- Implement an exemption process for policies. Plan for your workload teams to request exemptions from policies, and be ready to unblock the teams when needed.
- Use "Policy-driven-governance" to block workload teams when they attempt to deploy resources that don't meet a security control. This helps reduce configuration drift, where code declares a different state than what ends up being deployed.

Read more

- [Adopt policy-driven guardrails](#)
- [Bicep fundamentals](#)
- [Intermediate Bicep](#)
- [Advanced Bicep](#)
- [Use Bicep and GitHub Actions to deploy Azure resources](#)
- [Use Bicep and Azure Pipelines to deploy Azure resources](#)
- [Control and govern your Azure environment by deploying your infrastructure as code](#)

Subscription vending

Article • 06/18/2024

Subscription vending provides a platform mechanism for programmatically issuing subscriptions to application teams that need to deploy workloads. The following diagram shows where subscription vending fits in the platform and workload lifecycles.



Subscription vending builds on the concept of subscription democratization and applies it to application landing zones. With subscription democratization, subscriptions, not resource groups, are the primary units of workload management and scale. For more information, see:

- [Platform landing zones vs. application landing zones](#)
- [Democratized approach to subscriptions](#)
- [How many subscriptions should I use in Azure \(YouTube\)? ↗](#)

Why subscription vending?

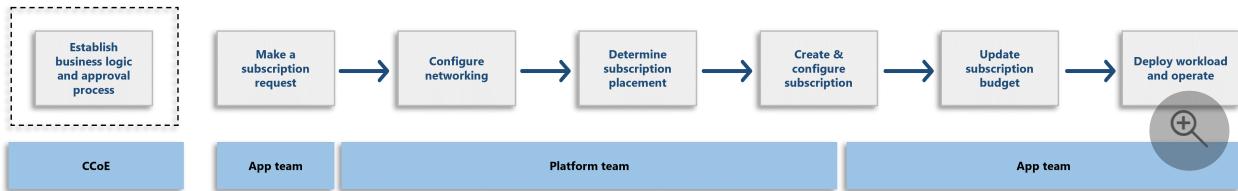
https://www.youtube-nocookie.com/embed/OoC_0afxAvg ↗

Subscription vending offers several benefits to organizations that need to deploy workloads in Azure. It standardizes and automates the process for requesting, deploying, and governing subscriptions for application landing zones. Subscription vending simplifies the subscription creation process and places it under the governance of the organization, so app teams can focus on deploying their workloads with greater confidence and efficiency.

- **Streamlined process:** Subscription vending provides an official front door for application teams to request subscriptions, eliminating the need for them to navigate the subscription process on their own.
- **Improved velocity:** Application teams can access application landing zones faster and onboard workloads quicker.
- **Efficient governance:** The platform team can enforce governance on application landing zones with minimal overhead.

How to implement subscription vending

Subscription vending involves three teams. The Cloud Center of Excellence (CCoE) establishes business logic and the approval process. When ready, the application teams make subscription requests. The platform team uses the request to create and configure the subscription before handing off the subscription to the application team. The application team updates the budget, deploys the workload, and establishes operations. The following guidance provides more details on each step of the subscription vending process. For more information, see [Subscription vending implementation guidance](#).



Platform teams can vend many options and subscription types to application teams. These types are referred to as *product lines* because they relate to platform-engineering principles and practices. For more information about choosing the option that best suits your needs, see [Common subscription vending product lines](#).

Establish business logic and approval process

To implement the subscription vending model, you need to establish an approval process that collects essential subscription information. The Cloud Center of Excellence (CCoE) should program the approval process and establish business rules around the information to collect.

Automate process. You should automate the process of subscription request capture and approval for faster provisioning and improved compliance.

Integrate with existing tooling. You should integrate the subscription vending approval process into your existing IT service management (ITSM) tool. The integration can simplify the approval process, reduce manual effort, and improve efficiency while reducing errors. It also makes maintenance easier over time and helps with compliance reporting for audits.

Connect to deployment pipeline. It's a best practice to tie the business logic of the approval process into the subscription deployment pipeline that the platform team manages. Azure Pipelines or GitHub Actions workflows are common solutions for the subscription deployment pipeline.

Gather requirements at intake. The business logic should allow application teams to request a subscription and provide subscription requirements. These requirements should include anticipated budgets, subscription owners, networking expectations, and

business criticality & confidentiality classification. Gathering this information at the beginning of the process informs your deployment parameters and stakeholder approval needs. The intake process should also give the platform team enough information to place the workload in the management group hierarchy.

With the approval process in place, application teams can start making subscription requests.

Make a subscription request

Subscription vending provides a standard process for application teams to request a subscription. It's important that you socialize the availability of subscription vending and ensure subscription requests are easy to make. After the application team submits a subscription request, the platform team assumes control of the process. The platform team maintains control until they create the subscription and deliver the subscription to the application team.

Configure networking

The subscription automation needs to set up the required networking components, and it needs to be flexible enough to meet the needs of each application team. As general guidance, never use overlapping IP addresses in a single routing domain. You can add or delete the address space of a virtual network without downtime if your size requirements change. For more information, see:

- [IP address restrictions](#)
- [Update address space of a peered virtual network](#)
- [Add or remove address range](#)

Use IP address management (IPAM) tool. You should use and integrate an IPAM system into the vending process to streamline IP address assignment. For more information and IPAM guidance, see [IP Address Management \(IPAM\) tools](#).

Grant the app team autonomy. You should grant application teams with the rights to create subnets and even some virtual networks in the subscription. The platform team should always create virtual networks that peer to a central hub.

Enforce networking governance. The platform team should enforce virtual network governance via (1) Azure policy assigned to the management group hierarchy or (2) Azure Virtual Network Manager and Security Admin Rules. For more information, see [Policy-driven governance](#) and [How to block high risk ports](#).

Determine subscription placement

The platform team should use the networking and governance requirements to place the subscription in the management group hierarchy. They should also review the subscription quota limits before creating the subscription. For more information, see [Tailor the Azure landing zone architecture to meet requirements](#).

Identify the right management group. Management groups help you organize and govern subscriptions and workload deployments. Locate or create a management group that enforces the policies needed for the classification and needs of each workload.

Build flexible automation. Your automation should be flexible enough (1) to deploy multiple subscriptions and (2) adapt to subscription service limits.

- *Multiple subscriptions:* Some workloads need several subscriptions. For example, some workloads have several instances separated by subscription. Alternatively, SaaS architectures that use dedicated resources per customer often use dozens of subscriptions.
- *Subscription service limits:* An enterprise with several thousand subscriptions should have automation that can deploy to an old subscription or colocate workloads in a subscription to avoid the limits. For more information, see [Azure landing zones FAQ](#).

You can request quota increases manually using the Azure portal after provisioning. It's easier if you automate this process by using the available APIs. However, the quota request can fail, so you should run a script to handle any errors. For more information, see [Microsoft.Capacity](#), [Microsoft.Quota](#), and [Microsoft.Support](#)

Create and configure subscription

You can now create and configure the requested subscription. The goal is to create a repeatable, consistent process. Automate as much of the subscription creation and configuration process as you can.

Use infrastructure as code (IaC). A common strategy for subscription vending is to create and configure the subscription programmatically by using IaC. You need a commercial agreement to create an Azure Subscription programmatically, but you can automate all aspects of subscription configuration without a commercial agreement. For more information, see:

- [EA required role](#)

- MCA required role(s)
- MPA required role

There are example subscription vending [Bicep](#) and [Terraform](#) modules to help you adopt a subscription vending model regardless of your enrollment in a commercial agreement. You should use GitHub actions or Azure Pipelines to orchestrate the automation.

Use tags for cost management. You should automate the consistent assignment of tags to each subscription for cost management and reporting purposes in Microsoft Cost Management. Although you receive billing reports with your commercial agreements, Cost Management provides greater functionality. For example, you can create reports for subscriptions with specific tags. For more information, see [How to use tags in cost and usage data](#) and [Group and allocate costs using tag inheritance](#)

Use production and non-production subscriptions. In the request for a new subscription, you must specify whether the workload is for Production or DevTest. DevTest environments result in lower resource charges but have other [terms](#). Note DevTest offer isn't available for MPA. For more information, see:

- [Azure billing offers and Active Directory tenants](#)
- [Resource organization design area overview](#)
- [Create Azure subscriptions programmatically](#)

Set up identity and role-based access controls (RBACs). Managing access to resources within an Azure subscription is critical for maintaining a secure and compliant environment. To control access, it's essential to set up identity and RBACs. This setup involves designating a subscription owner, creating Microsoft Entra groups to manage access, and establishing automation accounts to deploy workloads.

- *Designate a subscription owner.* The subscription vending automation needs to designate a subscription owner at creation. The subscription request should capture this information at intake. Subscription owners can only be users or service principals in the selected subscription directory. You can't select guest directory users. If you select a service principal, enter its App ID.
- *Create Microsoft Entra groups.* In addition to the subscription owner, you should ensure the vending process uses your Microsoft Entra group structure to manage access to the subscription. For elevated (for example, write) access, we recommend using [PIM for groups](#). Automating this creation process shouldn't violate best practices such as limiting the number of subscription owners and using the minimum required level of access.

- *Establish workload identities.* Workload identities (service principles) used for workload deployment often have elevated permissions at the subscription scope. The subscription request process should gather workload identity needs at intake. Your vending process should create these identities and assign appropriate subscription access. It's important to note that the workload identity can't use PIM and receives standing access to resources. We recommend you use managed identities to avoid the need to manage secrets. For more information, see [the identity design area](#).

Hand off to application team. After the platform team creates the subscription, they should hand off the subscription to the application team.

Update subscription budget

The platform and workload teams share responsibility for the financial health of the subscription. The deployment should create a subscription budget based on the information in the subscription request. The application should update the budget to meet their needs when they receive the subscription. Budgets are useful for auditing spending against current and forecast usage, but they aren't hard limits. You should create budget alerts to notify the subscription owners if the workload is about to exceed the budget threshold. For shared services, such as API Management, consider using [Azure cost allocation rules](#) to redistribute costs between consuming subscriptions.

Deploy workload and operate

The application team should have autonomy to create the resources they need for their workload and manage operations. The platform team remains responsible for subscription governance. As the governance requirements of a workload change, the platform team should move subscriptions to the management group that best meets workload needs. You can automate the move by using Bicep or Terraform. For more information, see:

- [Management groups overview](#)
- [Move subscription to new management group \(Bicep\)](#)
- [Move subscription to new management group \(Terraform\) ↗](#)
- [Tailor Azure landing zone to meet your requirements](#)

Next steps

Review the subscriptions, or product lines, that you can vend to application teams. Establish a great starting point so you can cater to a number of different scenarios.

Establish common subscription vending product lines

Feedback

Was this page helpful?

 Yes

 No

Establish common subscription vending product lines

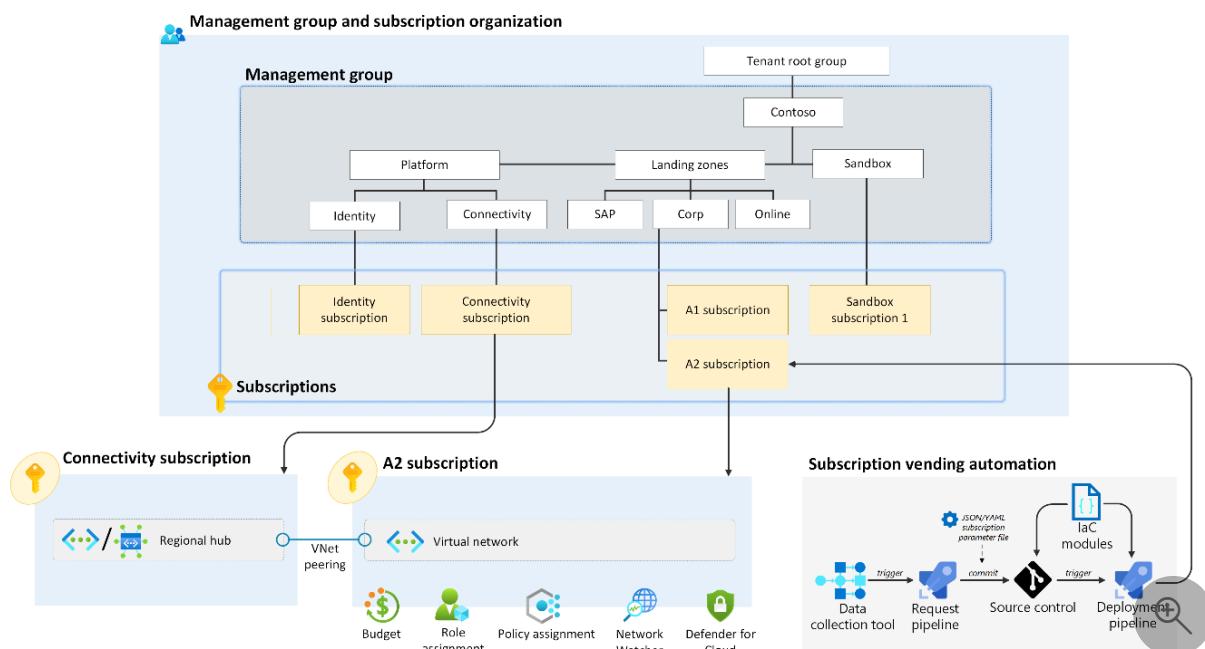
Article • 12/16/2024

Subscription vending helps organizations achieve the [subscription democratization design principles](#) of Azure landing zones, which is critical to consistent scaling, security, and governance of Azure environments. Subscription vending also helps organizations align with [platform engineering principles](#). For more information, see [Adopt a product mindset](#) and [Empower developers through self-service with guardrails](#).

Many organizations struggle to give their application teams the flexibility that they need to deliver their workloads and services effectively. One key obstacle is the lack of a standardized approach to [subscription vending](#), which can lead to confusion, delay, and inefficiency.

This article explores how platform teams can establish common subscription vending product lines that cater to the diverse needs of various application teams. The article discusses the benefits of offering various product lines and provides examples of common scenarios based on real customer deployments. You also learn why subscription vending doesn't have a "one size fits all" design and why you must provide various product lines to application teams.

The following diagram shows the organization of management groups and subscriptions within an Azure environment.



The following guidance describes why you might require various product lines and describes product line examples for customers that use Azure landing zones and subscription vending.

Take advantage of various product lines

Subscriptions that application teams require to deliver their workloads and services come in many types and styles. Outside of application teams, your organization might have other requirements that necessitate the use of an Azure subscription, such as various compliance and data-handling rules or architecture patterns.

When you decide on your organization's approach to designing and implementing subscription vending, consider asking these questions:

- What other resources should the platform team vend as part of the subscription vending process?
- For each application team, do you deploy multiple subscriptions, such as one per environment, by default?
- For every application, do you peer or connect the spoke virtual network back to your connectivity hubs by default?
- How should you structure role-based access control (RBAC) within each subscription?
- How should you govern and control resources and styles of architecture, or archetypes, that you use within subscriptions?

You can't address every application and platform team's unique requirements with any single subscription type or subscription style that you vend. Platform teams must give application teams flexibility to choose from multiple types and styles of subscriptions that the team can vend to them through a self-service system. These types of subscriptions are referred to as *product lines*.

Organizations that only provide a "one size fits all" approach to subscription vending often limit their internal customers' flexibility. For example, lack of flexibility might limit an application team's architecture design choices and potentially lead to compromises because of what they were vended.

Therefore, platform teams need to provide various product lines to cater to their organization's needs. This flexibility ensures that consumers can choose the product line that best fits their requirements.

Manage application environments

Your organization must manage application environments for application teams as part of your subscription vending processes and implementations. However, you should also provide flexibility so that application teams can manage their application environments, such as dev/test/prod, how they want to when they deliver applications. For more information, see [Environments, subscriptions, and management groups](#).

Some Azure services provide native features to help isolate an environment within a single resource instance in a single Azure subscription, such as Azure App Service with its deployment slots feature. This example forces application teams to use separate subscriptions, so teams can't take advantage of the full feature set of services that Azure provides. Separate subscriptions can also increase application delivery costs including operational and maintenance overhead.

Design common product lines for subscription vending

Now that you understand that platform teams must provide multiple Azure subscription types and styles, or product lines, to consumers of their Azure platform, this section describes several common product lines that you can use across industries and countries or regions.

Your platform team should use these common subscription vending product lines as a baseline. Your team can provide multiple options to its consumers out of the box, which aligns with the *prioritize customers* platform engineering principle. This approach gives internal customers the freedom to use Azure landing zone [design principles](#) and [design area recommendations](#) to deliver their workloads and service and also provides Azure platform governance.

Note

Use these examples as a starting point. You can customize and expand these product lines to cater to the needs of your organization.

Common product lines for subscription vending include:

- **Corp connected:** Workloads that require traditional Layer-3 IP routing connectivity to other applications and on-premises environments via the Connectivity subscription.

- **Online:** Workloads that connect with other applications through modern connectivity services and architectures, such as Azure Private Link or interaction via exposed APIs or endpoints from each application.
- **Tech platform:** Workloads that build a platform on which you can build other applications. For example, an Azure Kubernetes Service (AKS) fleet of clusters that an AKS platform team manages can host other applications within its AKS clusters on behalf of other application teams.
- **Shared application portfolio:** Shared workloads among the same application teams for a common set of closely coupled applications. You don't want to host the applications on their own or with any specific workload.
- **Sandbox:** An area where application teams can build a proof of concept (PoC) or minimum viable product (MVP) and impose fewer controls, so the team can promote development, invention, and freedom to build the best possible application from the catalog of available Azure services.

The corp connected product line

The *corp connected product line*, also referred to as an internal or private product line, for application landing zone subscription vending provides connectivity via traditional Layer-3 IP methods. You can use this product line to provide connectivity between resources that are:

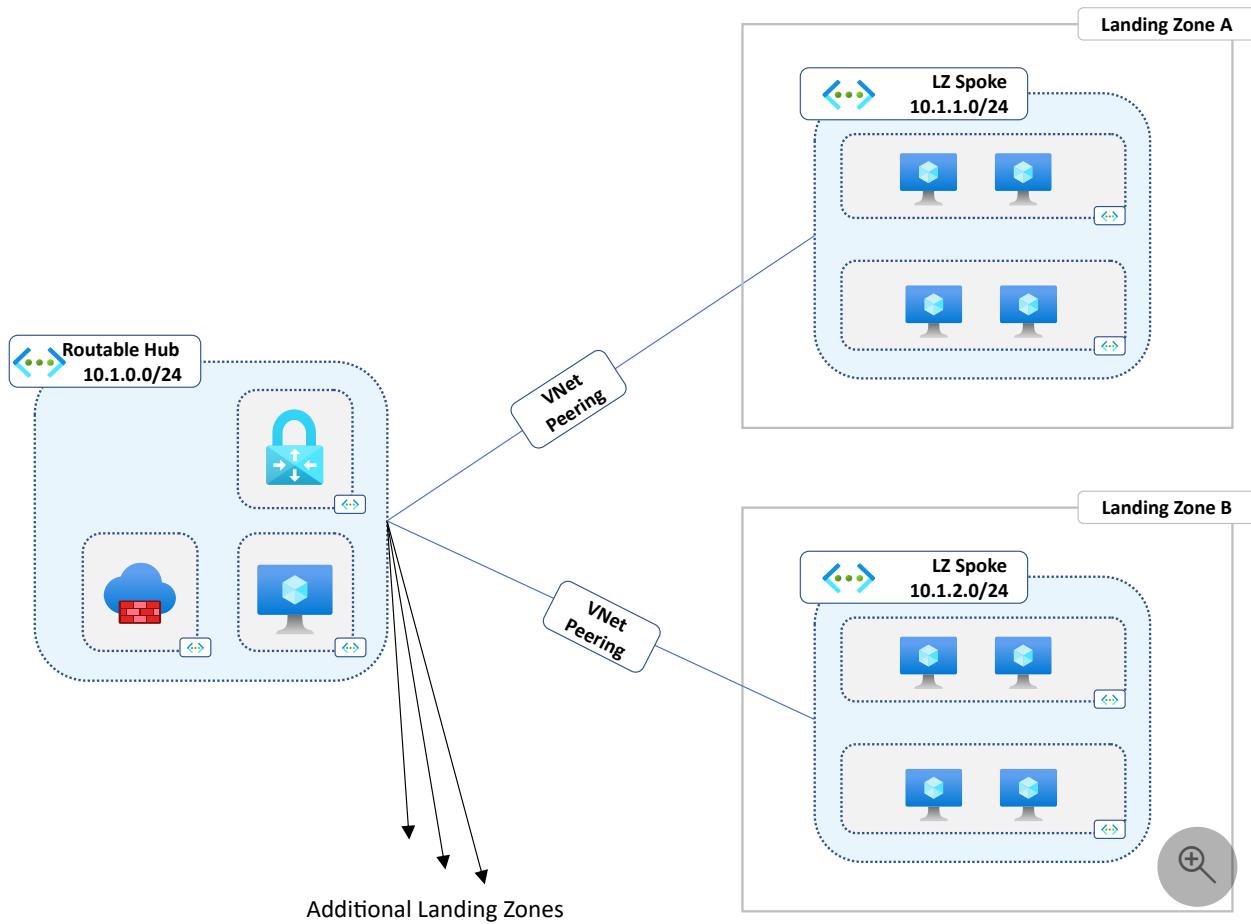
- In the same application landing zone.
- In different corp-connected application landing zones via an Azure firewall or network virtual appliance (NVA).
- On-premises or in different clouds via Azure ExpressRoute or VPN connections.

Organizations that use subscription vending often incorporate this product line because it aligns closely with how most on-premises environments work today. However, you should only use the corp connected product line when you need to. We recommend that you prefer more modern cloud-native approaches, such as the Online product line, when you can.

💡 Tip

For information about differences between corp and online workloads, see [What is the purpose of Connectivity, Corp, and Online management groups?](#).

The following diagram shows an example of the corp connected subscription vending product line. You can use this setup for a hub-and-spoke network model to help effectively manage network traffic and policies.



When to use the corp connected product line

Use the corp connected product line when:

- You want to perform Rehost and Refactor migrations and application builds based on [the five Rs of rationalization](#).
- You want to start your journey in Azure and are familiar with a similar on-premises architecture.
- You want to "lift and shift" applications into Azure.
- You want to enhance security between workloads by isolating applications into their own landing zone subscriptions and moving toward micro-segmentation principles from zero-trust without yet rearchitecting the application to be fully cloud-native.

Take note of these other considerations for the corp connected product line:

- Your platform team can vend the virtual network into the application landing zone subscription and peer the virtual network to the regional hub virtual network or the Azure Virtual WAN hub. Your team can then use an IP address management (IPAM) tool to control the IP address allocation.
- Platform teams don't usually vend subnets or any other resources into the virtual network. Instead, platform teams assign these activities to the application teams so that they can design their application networking how they want.
- Platform teams use an Azure policy that's assigned to the management groups above the subscription to enforce desired behavior, such as standardized network security groups (NSGs) attached to every subnet. The application team inherits this Azure policy and can't edit it. This approach follows the Azure landing zone design principle of subscription democratization.

The online product line

The *online product line*, also referred to as an *external or public product line*, for application landing zone subscription vending doesn't provide connectivity via traditional Layer-3 IP methods between resources in other application landing zones or on-premises through ExpressRoute or VPN connections. Resources in the same online application landing zone subscription can use virtual networks to communicate with each other via Layer-3 IP methods. But the virtual networks typically aren't peered back to regional connectivity hubs or other application landing zones.

Instead, you can provide connectivity via public interfaces between resources that are:

- In different application landing zones.
- On-premises.
- In workloads that are in different clouds.

You can secure the connections with network controls, authentication features, and authorization features that are exposed by the various platform as a service (PaaS) solutions that you use to construct the application.

You can use the [Private Link service](#) and [Azure private endpoints](#) inside and between online application landing zone subscriptions to enable and expose private, Layer 3-based connectivity between applications. You can also use this approach between the PaaS services that you use within application landing zones to prevent the use of these PaaS services' public interfaces for security or regulatory control.

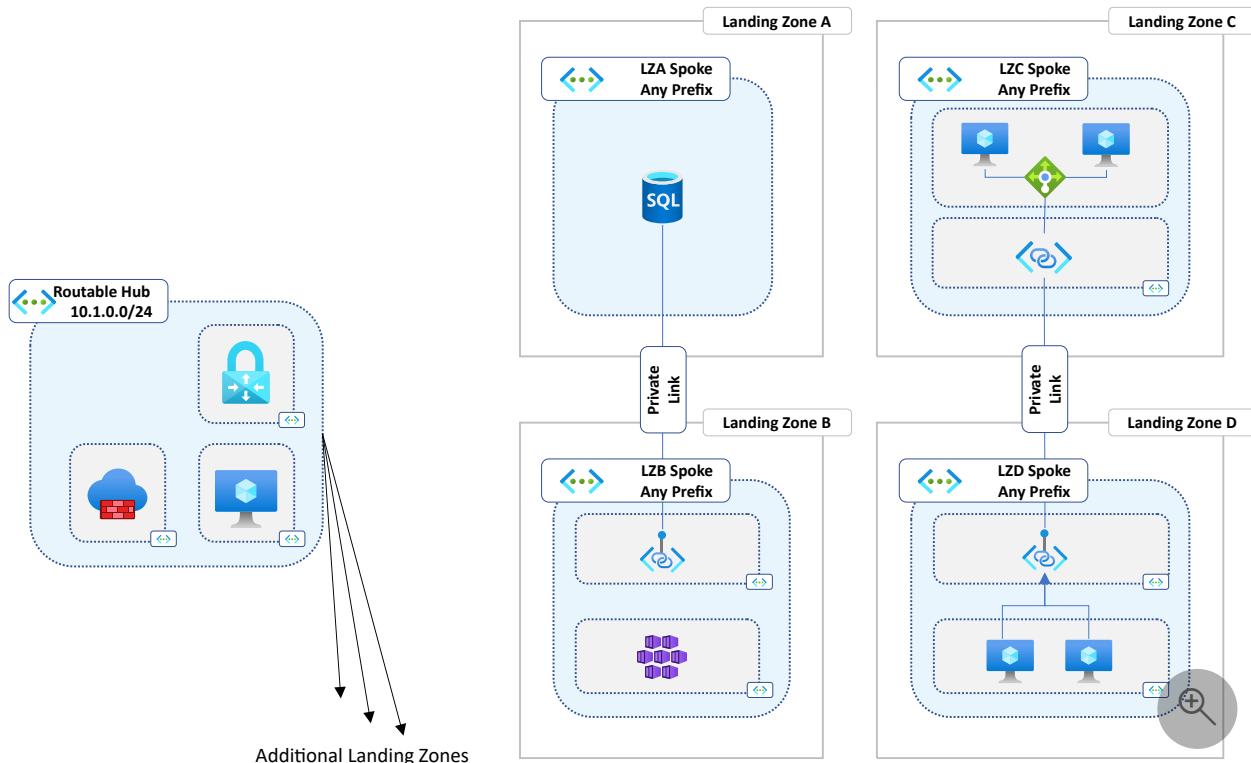
You can also use the [Private Link service](#) with [private endpoints](#) to expose and publish applications that you host within online application landing zones to corp connected application landing zones, on-premises locations, or other clouds. You can place private endpoints in either corp-connected application landing zones or directly in connectivity hubs, which then grant access to these private endpoints via traditional Layer-3 connectivity methods such as virtual network peering, ExpressRoute connections, or VPN connections.

Think of the online application landing zone product line as isolated islands. By default, the only resources that can access resources within the subscription are the resources that you deploy within the same online application landing zone subscription. As mentioned previously, you can then use the techniques in this article to expand connectivity to other application landing zones, on-premises locations, or other clouds.

💡 Tip

For more information about the differences between corp and online workloads, see [What is the purpose of Connectivity, Corp, and Online management groups?](#).

The following diagram shows an example of an online subscription vending product line.



When to use the online product line

Use the online product line when you want to:

- Refactor, rearchitect, rebuild, and perform migrations and application builds, based on [the five Rs of rationalization](#).
- Provide application teams with a fully democratized application landing zone to use, even regarding networking configuration.
- Take advantage of cloud-native services and architectures.
- Considerably enhance alignment with zero-trust principles.
- Use the corp connected product line, but private IP address space is unavailable or limited.
 - In this scenario, you should review the guidance in [Prevent IPv4 exhaustion in Azure](#).

The Tech platform product line

Teams that use technology platforms, such as Azure VMware Solution or Azure Virtual Desktop, should implement the *tech platform product line*. The tech platform product line is essentially a subscription vending product line that better suits highly technical requirements. You can use the tech platform product line to host and manage large and complex workloads that typically host multiple applications for several other application teams across your organization. Use this product line if your application team manages only the application parts and not the underlying technology platform pieces.

💡 Tip

To better understand this product line, consider the following example. A technology platform team, like an AKS team, aims to offer AKS as a managed service to other application teams that need to run their applications on the AKS platform. The AKS tech platform team provides the management, maintenance, security, and configuration of AKS. So the application team only maintains their application and deploys it on the platform.

You might include the following products in a tech platform product line:

- An **App Service Environment**, typically via separate App Service plans.
- **AKS**, typically via namespaces within one or more clusters.
- **Azure Virtual Machines** on Azure VMware Solution clusters or hosts.

- Azure Virtual Desktop to provide virtual desktops or applications to your entire organization.

You can include these products in either [corp connected](#) or [online](#) product lines, depending on the requirements for the technology platform that your team wants to provide as a service to other application teams in your organization.

Shared application portfolio

The *shared application portfolio product line* for application landing zone subscription vending is for workloads that don't need several separate application landing zone subscriptions for simple applications that might be constructed from only a small number of Azure resources.

Your application teams and departments can use this product line to host several small applications or shared components, such as storage accounts or SQL servers. The teams share these components between several of their own applications in a single subscription or a small number of subscriptions.

ⓘ Important

A common team owns subscriptions that you vend under this product line. This team manages the related portfolio of applications that you deploy in this subscription for this product line. Don't use this product line for general deployments of unrelated application workloads that have distinct application portfolio owners.

Plan carefully to ensure ongoing flexibility, access control, governance, and maintainability if your organization changes to a single subscription and uses resource groups to delegate access.

If you consider resource group delegation in a single subscription between multiple teams, factor in the following considerations before you make a final decision:

ⓘ [Expand table](#)

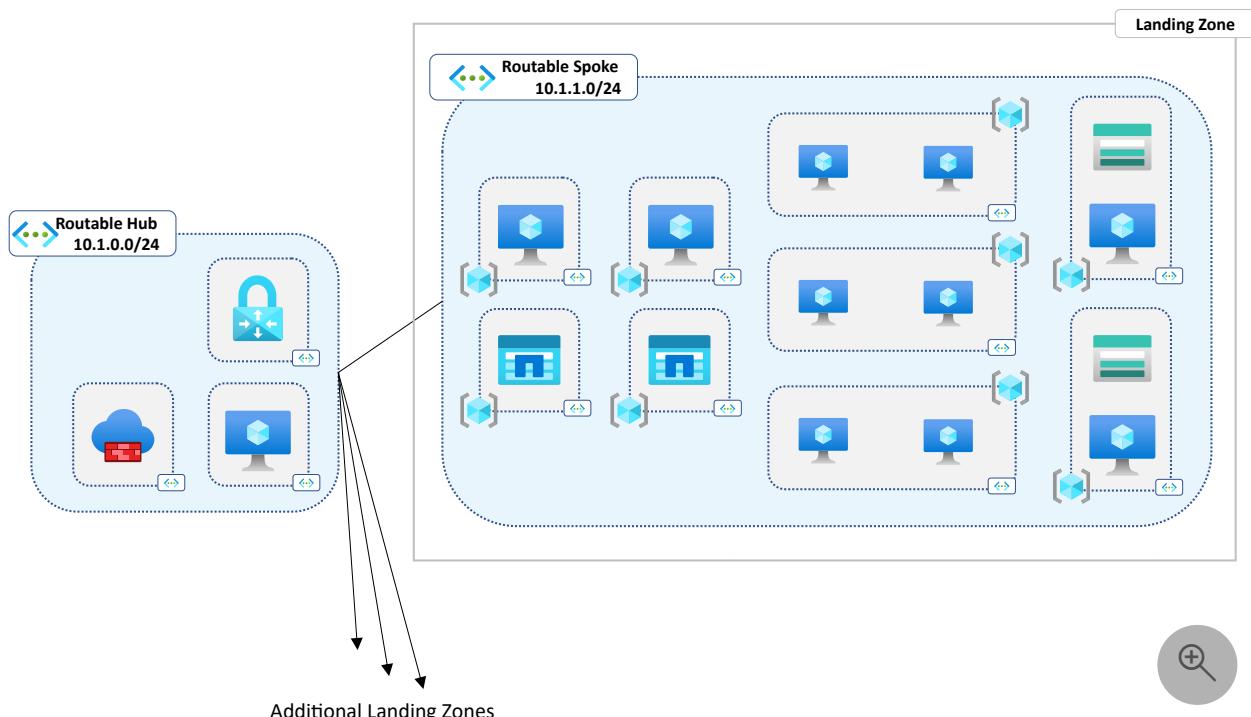
Area	Considerations
Common ownership of related application portfolio	- Have a common owner, such as a business unit of a department, manage applications to simplify change management so that it remains within the approval scope of the same entity.

Area	Considerations
	<ul style="list-style-type: none"> - Ensure that workloads follow consistent policy assignment across the subscription, including logging, monitoring, and security.
Regulatory compliance	<ul style="list-style-type: none"> - Use IAM and Azure policies to create subscriptions for workloads that have regulatory compliance requirements, including National Institute of Standards and Technology (NIST), Center for Internet Security (CIS), Payment Card Industry Security Standards Council (PCI SSC), industry requirements, and regional requirements. For more information, see Tailor Azure landing zones. - Create subscriptions for workloads that use privacy and data-handling requirements for governance. Individual subscriptions reduce access.
Azure Policy	<p>Scope Azure policies to management groups, subscriptions, resource groups, and resources. Assign Azure policies at a high scope level for efficient governance when you deploy resources in resource groups.</p> <p>Consider the following constraints when you manage Azure Policy at the resource group scope level:</p> <ul style="list-style-type: none"> - Increases management overhead to create Azure Policy assignments when you add new resource groups to subscriptions - Increases workload when you manage changes to policy assignments - Increases security and governance gaps when you don't immediately assign policies to resource groups - Reduces the ability to roll up compliance status at high scopes, such as management groups and subscriptions
Subscription limits	<ul style="list-style-type: none"> - Check limits to ensure that applications don't hit hard limits that prevent growth. Each subscription has soft and hard limits for Azure services. - Create separate subscriptions for applications that anticipate large growth patterns that meet subscription limits. - Don't share subscriptions with application teams from different business units or departments to prevent noisy neighbor problems.
Azure services and feature alignment	<p>You can deploy services that provide basic Azure service primitives, such as Virtual Machines, virtual networks, and simple PaaS services, within a single resource group. But the complexity of modern composite offerings can require that you deploy these more complex services outside the boundaries of a single resource group. Use other democratized subscription approaches that are described earlier in this article for these deployment scenarios.</p>

Area	Considerations
Only platform teams can create resource groups	<p>When you share a subscription among various application teams across business units or departments, you might restrict any team's ability to create new resource groups in the shared subscription.</p> <p>This restriction limits resource group sprawl. Only the platform team can create and govern new resource groups.</p> <p>This approach increases the complexity of RBAC assignments and places an increased dependency on platform teams to manage application deployments, which can impede application teams' agility and empowerment.</p>

You can place the subscriptions that you vend in the shared application portfolio product line under either Corp or Online management groups. This method aligns with the Azure landing zones default recommended hierarchy. Alternatively, you might place the subscriptions beneath new management groups if your organization's management group hierarchy follows the guidance in [Tailor the Azure landing zone architecture to meet requirements](#).

The following diagram shows an example of the shared application portfolio subscription vending product line.



Use the shared application portfolio product line if:

- Your application team needs to deliver several small resources or components that their applications share, but the components don't directly fit in any of the dedicated application landing zones.

- You have resources or components that you need to share between applications in the same department, but the components don't directly fit in any of the dedicated application landing zones.
- Technology platform teams want to host large, shared services that are managed, such as AKS, Azure Virtual Desktop, and Azure VMware Solution, so that other application teams can use or host their applications on the services.

Sandbox

Use the *sandbox product line* for application landing zone subscription vending to help provide safe, lightly governed, and visible testing areas to build PoCs or MVPs in Azure.

For more information, see [Landing zone sandbox environments](#) and [Manage application development environments in Azure landing zones](#).

Sandboxes are often time-boxed or budget-constrained, which means that they have a time limit or budget limit. In these cases, you must either extend or remove and decommission the sandbox.

If your organization doesn't provide a sandbox product line for application teams or others to test and experiment with services in Azure, teams might resort to *shadow IT* setups. If so, your organization might struggle to provide reporting and visibility and apply governance to subscriptions that business users create outside of your platform team's control and oversight.

Your platform team must provide easily accessible, preferably self-service, and automatically approved access to sandbox subscriptions for your organization's users and teams. Provide users and teams access to an environment that your platform team can view and govern to prevent *shadow IT* environments that the platform team can't access or control, which creates risk.

Sandboxes often follow the networking configuration approach of online product line subscriptions because you don't peer them to other virtual networks outside of the sandbox's subscription boundary. Sandboxes also often have extra controls to prevent hybrid connectivity to on-premises locations or other locations. Use these controls so that unknown sources can't exfiltrate data from sandboxes to unapproved locations. You can use an Azure policy to enforce these controls.

Just like the shared application portfolio and tech platform product lines, you can also share the sandbox product line among teams in the same department with the same considerations. Don't create a single sandbox subscription and share it among teams via resource groups. Instead, create additional sandbox subscriptions.

Use the sandbox product line if you need to provide a safe, secure, and governed Azure subscription to anyone across your organization who wants to experiment, create PoCs, or create MVPs in Azure. You must lightly govern these users and grant them access to all services to prevent *shadow IT* practices.

Summary and takeaways

This article outlines prescriptive guidance to help you navigate complex subscription vending processes and move toward implementation.

Determine the requirements of your future application teams to choose the subscription vending product line that best suits them. Identify requirements for the initial set of workloads that you build or migrate to help prioritize the subscription vending product lines that you want to enable and expose via a self-service interface.

Each product line has an implementation cost and a maintenance cost. Evaluate the long-term cost versus long-term benefits and usage.

Customers typically enable the following subscription vending product lines initially:

- [Sandbox](#)
- [Corp connected](#)
- [Online](#)

Additional resources

To further support your platform-engineering approach, review the following resources when you design and implement your organization's subscription vending product lines and offerings:

- [Video: How many subscriptions should I use in Azure? ↗](#)
- [Platform landing zones vs. application landing zones](#)
- [Policies included in Azure landing zones reference implementations ↗](#)
- [Tailor the Azure landing zone architecture to meet requirements](#)
- [What is the purpose of Connectivity, Corp, and Online management groups?](#)
- [Manage application development environments in Azure landing zones](#)
- [Platform engineering principles](#)

Next step

For the best results, you should automate as much of the subscription vending process as possible. Use the companion guidance about implementing subscription vending

automation.

Subscription vending implementation guidance

Feedback

Was this page helpful?

 Yes

 No

DevOps considerations

Article • 05/31/2024

This article provides considerations and recommendations for DevOps in Azure landing zones.

What is DevOps

DevOps is the union of people, processes, and technology that provides continuous value to development (dev) and operations (ops). The DevOps approach encourages team collaboration that creates repeatable processes to help organizations operate efficiently and at scale.

In the context of Azure landing zones, DevOps becomes the framework that guides your team (or teams) responsible for your entire Azure landing zones lifecycle management in areas such as:

- How to self-organize and define boundaries with other teams to achieve the appropriate balance between autonomy and governance
- How to continuously evolve Azure landing zone architecture design ([Conway's Law](#))
- How to plan, prioritize and iterate the implementation of the designed architecture
- How to implement version control, continuous integration and continuous deployment for Azure landing zone code
- How to operate and respond to incidents for systems and platforms you own
- The level of automation you apply to Azure landing zone provisioning and self-healing
- How to collaborate with other teams in your organization in an agile, outcome-oriented way
- How to create a generative culture of security, quality, user-centricity and continuous learning

The decisions you make when reviewing cloud operating models can influence how you use your DevOps framework.

DevOps design considerations

- Define your [DevOps framework](#), or align it with your organizational's DevOps and cloud adoption strategy. Include the definition of DevOps and the principles and

practices your team must follow. Make sure you connect your DevOps strategy to your business strategy.

- Establish [metrics](#) that allow your team to improve their DevOps performance. High-performance teams use a hypothesis to test their ideas, measure it to see how the hypothesis works, then make changes as needed. DevOps' final intent is to improve aspects like deployment frequency, mean time to apply a change, or time to restore a degraded service. You must design all these metrics to eventually affect overall business performance.
- Determine the [DevOps practices](#) your team should implement first based on their current skilling, and design a roadmap to incrementally apply new practices that help your team improve their DevOps metrics. Investing in engineering capabilities and resources is critical.
- Determine the [DevOps toolchain](#) your team should use to implement the DevOps practices. Make sure that the tools are consistent with your overall DevOps strategy to avoid scenarios of heterogenous DevOps ecosystems increasing the complexity of Azure landing zone or workload deployments.
- Evaluate the effect that your implemented DevOps practices and DevOps tools have on the design of your Azure Landing Zones.
- Create a readiness plan to continuously grow your team's skills. Blanket application of a DevOps model doesn't instantly establish capable DevOps teams.
- [Determine the team topology](#) that best aligns with your organization's DevOps strategy and cloud operating model, and establish clear boundaries, responsibilities and dependencies between the teams.
- Determine how the team responsible for Azure Landing Zones should collaborate with other teams in your organization to capture new Azure Landing Zone requirements to update design and implementation, resolve incidents, minimize dependencies, and align with business priorities.

DevOps recommendations

The following sections contain recommendations to help you implement the DevOps framework within your organization.

Define your DevOps framework

To establish your DevOps framework, consider using the frameworks that are already available to start with a set of predefined proven practices:

- [Microsoft DevOps Resource Center](#) provides a rich set of definitions, practices, and capabilities that you can adapt to Azure Landing Zone lifecycle management, including:
 - [Planning](#)
 - [Development and Continuous Integration](#)
 - [Continuous Delivery](#)
 - [Operations and Reliability](#)
 - [DevSecOps](#)
- [Microsoft DevOps Dojo](#) establishes a DevOps taxonomy built on four foundational pillars and eight capabilities:
 - Pillars:
 - [Culture](#)
 - [Lean Product](#)
 - [Architecture](#)
 - [Technology](#)
 - Capabilities:
 - [Continuous Planning](#)
 - [Continuous Integration](#)
 - [Continuous Delivery](#)
 - [Continuous Operations](#)
 - [Continuous Security](#)
 - [Continuous Quality](#)
 - [Continuous Improvement](#)
 - [Continuous Collaboration](#)

Define DevOps practices for your Azure landing zones management

Consider the following DevOps practices for your Azure landing zones:

- Review how to [manage the development lifecycle of Azure landing zones as code](#).
- Review [security considerations](#) for Azure Landing Zones in the DevOps space.

Plan your DevOps implementation journey

Define and align your DevOps implementation journey with your organization's [cloud adoption plan](#).

- Determine where your team is today in the following areas:
 - DevOps practices your team has adopted for Azure Landing Zones management.
 - Use tools like the [DevOps Capability Assessment](#) to assess the current state of your team's DevOps status.
 - Current team [structure](#), including [roles and responsibilities](#) and owned [cloud functions](#).
 - What [technical skills](#) your team has.
 - Which [cloud operations model](#) your team currently follows.
- Use your organization's cloud adoption plan to define a desired model for your team.
- Establish an iterative roadmap for implementing the desired model in an iterative and incremental mode that aligns with your organization's [transformation timeline](#).

Implement desired DevOps metrics

Identify which metrics you'll use to measure your team's DevOps performance. Use metrics to drive desired habits in your team that connect with [business outcomes](#). Establish metrics to let your team measure impact over activities. Make key metrics visible to all, since transparency promotes trust and drives alignment with organizational objectives.

Examples of metrics that measure DevOps performance to improve business impact include:

- Business Outcomes:
 - Use [Objectives and Key Results](#) as a tool to move your teams away from an "output" mindset and toward an "outcome" mindset. For example, you might use the number of workloads that improved their compliance rating over the number of policies deployed to Azure.
 - Customer or end user satisfaction. Examples include Net Promoter Score (NPS), surveys, interviews.
 - Business growth. Examples include increased profitability, increased revenue, and new revenue source requirements.
 - People metrics. Examples include the Employee Net Promoter Score (eNPS), utilization, retention, and satisfaction.
 - Costs. For example, you might use reduction in costs.
- Software delivery performance:
 - Lead Time for Change, the time it takes for a bug fix, new feature or any other change to go from idea to deployment to production.

- Deployment Frequency, the deployments per day of code changes to production.
- Mean Time to Restore, the time it takes to restore service in production after an incident occurs.
- Change Fail Percentage, the percentage of changes to production (such as configuration changes) that lead to a failure.
- Quality:
 - Defect escape rate, the number of defects identified by your end users.
 - Unplanned work or rework, the percentage of time spent doing unplanned work or rework.
 - Active bugs, the number of bugs that aren't yet fixed.
 - Code Health, the percentage of code that hasn't been unit tested.

Define your DevOps technology ecosystem

The DevOps toolchain you choose to manage the lifecycle of your Azure Landing Zones affects:

- Your strategies for implementing DevOps principles and practices
- Security considerations for your DevOps lifecycle
- The overall architecture design of your Azure Landing Zones lifecycle management

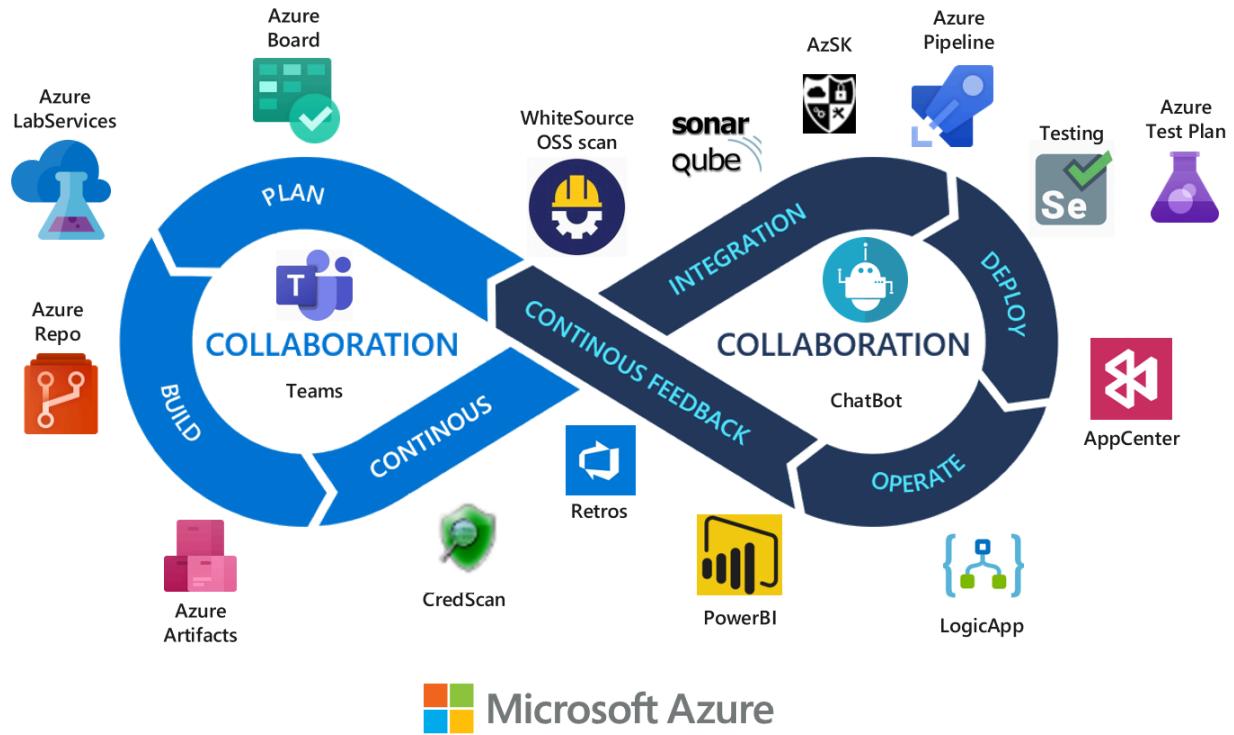
Use the [DevOps framework](#) that you previously defined to identify which tools to use for each of your DevOps processes. Choose the DevOps technologies that are most suitable for your teams' needs, but find a balance that lets you achieve standardization across your organization but avoid too much complexity or heterogeneity in your DevOps ecosystems.

Examples of DevOps technologies across different DevOps stages include:

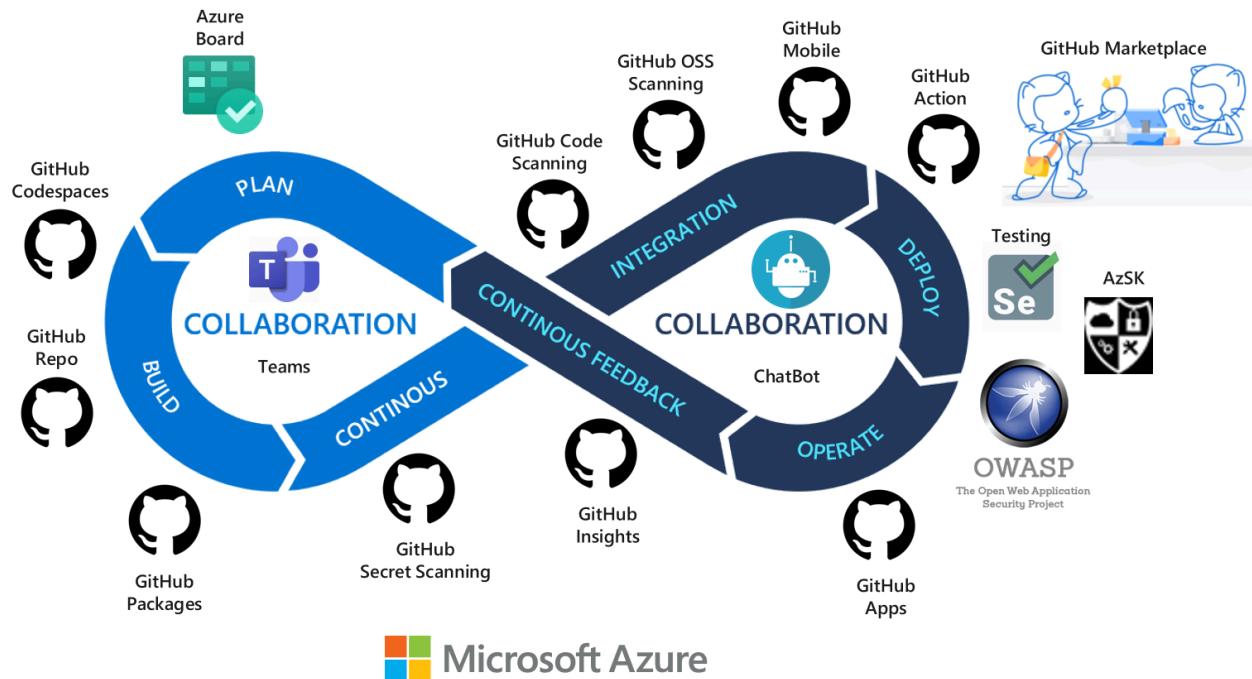
- **Planning:** Atlassian Jira, Atlassian Trello, Azure Boards, GitHub
- **Continuous integration (CI) and testing:** Atlassian Bitbucket, Azure Repos, GitHub Repos, npm, NuGet, Selenium, SmartBear Cucumber, SonarSource SonarQube, Zed Attack Proxy
- **Continuous delivery (CD):** Atlassian Bamboo, Azure Pipelines, GitHub Actions, Jenkins, Octopus Deploy, Perforce Puppet, RedHat Ansible
 - **Infrastructure as code:** Bicep, Pulumi, Terraform
 - **Bootstrapping:** ArgoCD GitOps, Flux GitOps, Progress Chef, PowerShell Desired State Configuration (DSC)
- **Operations:** Azure Automation, Azure Monitor, CISCO Splunk, Grafana, Microsoft Power BI

- **Collaboration and feedback:** Atlassian Confluence, Azure DevOps Wikis, GitHub Discussions, GitHub Wikis, Microsoft Teams, Slack, Stack Overflow

The following diagram shows an example DevOps framework with Azure DevOps toolchain selection:



The following diagram shows an example DevOps framework with Azure DevOps and GitHub toolchain selection:



Next step

Feedback

Was this page helpful?

 Yes

 No

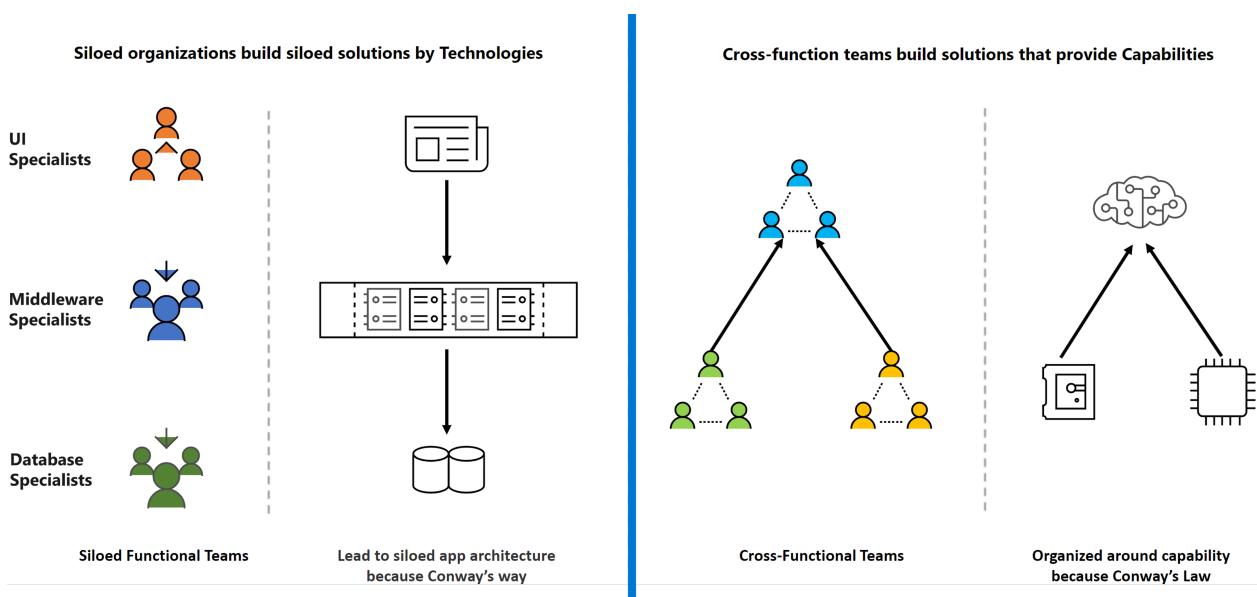
DevOps teams topologies

Article • 03/27/2023

The distribution of roles, responsibilities, and trust between IT teams and applications teams is paramount to the operational transformation involved in cloud adoption at scale.

IT teams strive to maintain control. Application owners seek to maximize agility. The balance you ultimately establish between these two goals greatly influences the success of your cloud operating model.

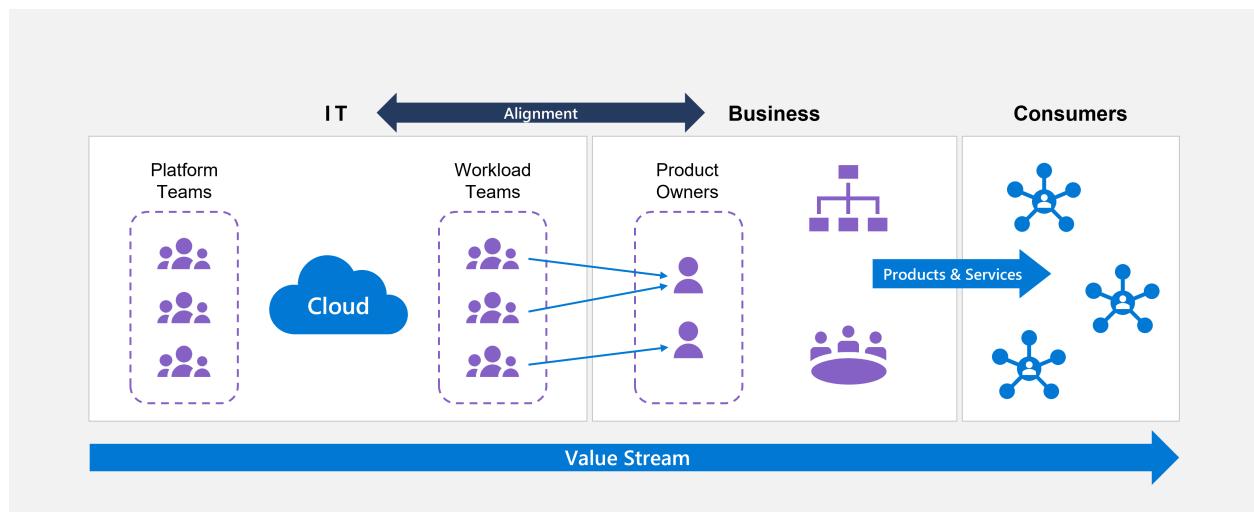
According to Conway's law, teams produce Architectures based on their communication structure. Understanding this principle is critical as you work to achieve the necessary balance between autonomy and control. Any organization that designs a system (defined broadly) will produce a design structure that's a copy of that organization's communication structure.



From a DevOps perspective, organizations must optimize for rapid response to customer needs. Teams that own, design, and implement their applications and systems find their highest level of autonomy in architectures with the following characteristics:

- Evolutionary architecture that supports constant changes
- Deployability
- Testability

Conway's solution is to outmaneuver Conway's Law. If your organization follows a particular structure to produce services and products and is looking to optimize, you need to rethink your organizational structure. Evolve your team and organizational structure to achieve your desired architecture.



This principle leads to intentionally designed [team topologies](#) in which teams are responsible for the end-to-end of any applications, systems, or platforms they own in order to achieve the full discipline of DevOps.

The following table provides a simplified categorization of these teams.

Team type	Definition
Application workload teams	These teams build applications that drive direct business outcomes for a segment of the business domain. In the context of Azure Landing Zones, these teams are responsible for the end-to-end lifecycle of application workloads.
Platform teams	These teams build compelling internal platforms to accelerate delivery and reduce the cognitive load of application workload teams. In the context of Azure Landing Zones, these teams are responsible for the end-to-end lifecycle of the Azure Landing Zone.
Enabling teams	These teams help overcome skill gaps by assisting other teams with specialized capabilities like DevOps.

Design considerations

- Establish a cross-functional platform team to design, build, provision, manage, and maintain your Azure Landing Zone lifecycle. This team can include members from your central IT team, security, compliance, and business units to ensure that a wide spectrum of your enterprise is represented. Make sure you avoid [antipatterns](#).
- Consider establishing an enabling team that can provide DevOps functions to support applications and platforms that don't have existing DevOps capabilities, or a business case to establish one (for example, legacy applications with minimal development capabilities).

- Don't restrict your application workload teams to central artifacts, since it can hinder their agility. You can use policy-driven governance and Azure role-based access control (Azure RBAC) to enforce consistent baseline configurations and ensure that application (business unit) teams are flexible enough to innovate yet still able to draw from a predefined set of templates.
- Don't force your application teams to use a central process or provisioning pipeline for the application resource instantiation or management. Existing teams that already rely on a DevOps pipeline for application delivery can still use their current tools. Remember, you can use [Azure Policy](#) helps to enforce organizational standards and to assess compliance at-scale and address [security considerations](#) for your DevOps processes.
- Blanket application of a DevOps model doesn't instantly establish capable DevOps teams.
- Investment in engineering capabilities and resources is critical.
- Application teams for some legacy applications might not have the engineering resources required to align with a DevOps strategy.

Design recommendations

The following sections contain design recommendations to guide you as you design your team topologies.

Align team topologies with your cloud operating model

Make sure you align your team topologies with your desired [cloud operating model](#).

Establish a core process for [operational fitness reviews](#) so you fully understand the problems that can result from your team structures.

Define functions for your platform team

The following list provides a recommended set of functions for the platform team responsible for your Azure Landing Zones:

- Architecture governance
- Subscription provisioning and delegation of required network, identity and access management policies
- Platform management and monitoring (holistic)

- Cost management (holistic)
- Platform-as-code (management of templates, scripts, and other assets)
- Overall operations on Microsoft Azure within your Azure Active Directory tenant (management of service principals, Microsoft Graph API registration, and role definitions)
- Azure RBAC (holistic)
- Key management for central services (simple mail-transfer protocol and domain controllers)
- Policy management and enforcement (holistic)
- Security monitoring and audits (holistic)
- Network management (holistic)

Define functions for your application workload teams

The following list provides a recommended set of functions for your application teams responsible for application workloads:

- Creation and management of application resources through a DevOps model
- Database management
- Application migration or transformation
- Application management and monitoring (application resources)
- Azure RBAC (application resources)
- Security monitoring and audits (application resources)
- Secrets and Keys management (application keys)
- Cost management (application resources)
- Network management (application resources)

Define functions for enabling teams

The following list provides a recommended set of functions for an enabling team responsible for assisting your other teams:

- Definition of horizontal (cross-function) guidance and capabilities to help acquire the right expertise across your organization, which ensures alignment with your overall target cloud operating model (like DevOps)
- Support, training and coaching for other teams to reach the necessary level of expertise
- Establishment of a common set of reusable templates and libraries for your application or platform teams, and fostering InnerSourcing, such as [Common Azure Resource Modules Library](#).

Define interaction modes between teams

The goals of interactions between your teams are to:

- Achieve autonomy
- Unblock dependencies
- Minimize waste time
- Avoid bottlenecks

[Team Topologies](#) outlines three team interaction modes:

Interaction mode	Description
Collaboration	Teams work closely together.
X-as-a-Service	Teams consume or provide something to other teams with minimum collaboration, similar to third-party vendor interactions.
Facilitating	Teams help or are helped by another team to remove impediments.

Development lifecycle

Article • 10/14/2024

Development lifecycle strategy provides key design considerations and recommendations for repository, branch, automated builds, deployment, and rollback strategy during automatic landing zone creation.

Repository strategy

Design considerations

- Consider adopting a version control system like Git to provide your team with flexibility in code sharing and management.
 - [Git](#) is the industry-standard version control system. It's a distributed version control system, where your local copy of the code is a complete version of the repository.
- Understand mono-repo versus multirepo [Repository structure](#).
 - In mono-repo structures, all source code lives in a single repository.
 - In multirepo structures, all projects are organized into separate repositories.
- Choose a visibility setting that suits the content of your repository.
 - Public repositories can be accessed anonymously.
 - Private repositories require users be granted access to the repo and signed in to access services.
 - You can set public and private visibility for [Azure DevOps Projects](#) and [GitHub repos](#).
- Consider setting repository permissions that help you control who can contribute to your source code and manage other features.
 - You can set repository permissions for [Azure DevOps](#) and [GitHub](#).
- Consider using [Infrastructure as Code \(IaC\)](#) resource deployment to Azure. IaC allows you to manage infrastructure in a declarative model, helping to reduce configuration effort, ensure consistency between deployments, and avoid manual environment configuration.
- Azure provides support for IaC for Landing Zones through:
 - [Bicep](#)
 - [Terraform](#)

Design recommendations

- Use Git as a version control system.
- Use private repositories when building Azure Landing Zones
- Use public repositories when sharing nonconfidential information like automation examples, public documentation, and open-source collaboration material.
- Adopt an IaC approach for deploying, managing, governing, and supporting cloud resources.

Branch strategy

Design considerations

- Consider using a [branch strategy](#) that allows teams to collaborate better and efficiently manage version control.
- Consider using specific [naming conventions](#) for your branches.
- Consider using [branch permissions](#) to control user capabilities.
- Consider using branch policies to help your teams protect important branches of development. Policies that can help enforce code quality and change management standards. Examples of branch policies include:
 - Always using pull requests to merge changes into important branches.
 - [Requiring a minimum number of reviewers](#) for pull requests.
 - [Automatically including code reviewers](#).
 - [Checking for linked work items](#) allows you to keep traceability.
 - [Checking for comment resolution](#) validates whether all PR comments are resolved.
 - [Limiting merge types](#).
- Adopting a pull request strategy can help you keep control of code changes merged into branches.
 - Define a [merge strategy](#).
 - Pull requests should be simple, with the number of files kept to a minimum to help reviewers validate commits and changes more efficiently.
 - Pull requests should have clear titles and descriptions so reviewers know what to expect when reviewing code.
 - You can use [pull request templates](#).

- You can delete origin branches after pull requests are complete, which gives you more control and better branch management.

Design recommendations

- Adopt a [trunk-based development model](#), in which developers commit to a single branch. This model facilitates continuous integration. All feature work is done in the trunk, and any merge conflicts are resolved when the commit happens.
- Have your teams define and use consistent naming conventions for branches to identify the work done.
- Set permissions to control who can read and update code in a branch of your Git repository. You can set permissions for individual users and for groups.
- Set branch policies:
 - Require the use of pull requests for branch merges into the main branch.
 - Require a minimum number of reviewers for pull requests.
 - Reset all approval votes to remove all approval votes, but keep votes to reject or wait whenever a source branch changes.
 - Automatically include code reviewers.
 - Check for comment resolution.
- Set squash as merge strategy, which allows you to condense the Git history of topic branches when you complete pull requests. Instead of adding each commit on a topic branch to the history of the default branch, a squash merge adds all file changes to a single new commit on the default branch.

Automated builds

Design considerations

- Consider implementing [Continuous Integration \(CI\)](#). CI involves merging all developer code into a central codebase on a regular schedule and automatically executing standard builds and test processes.
- Consider using CI triggers:
 - [Azure Repos Git](#). You can configure branches, paths, and tags as triggers to run a CI build.
 - [GitHub](#). You can configure branches, paths, and tags triggers to run a CI build.
- Consider including IaC unit tests in your build process to validate syntax.

- The [ARM Templates test toolkit](#) checks whether a template follows recommended practices.
- [Bicep linter](#) checks Bicep files for syntax errors and best practice violations.
- Consider including unit tests in your application build process. Review the tasks available for [Azure DevOps Pipeline](#).
- Use Azure DevOps service connections or GitHub secrets to manage connections to Azure. Each connection should have the correct privilege access to Azure resources.
- Consider using [Azure Key Vault secrets](#) to store and manage sensitive information like passwords, API keys, certificates.
- [Azure DevOps agents](#) can be self-hosted or Microsoft-hosted.
 - Maintenance and upgrades are taken care of for you when you use Microsoft-hosted agents. Every time a build job is run, a fresh virtual machine is created.
 - You set up and manage self-hosted agents on your own to run build jobs.

Design recommendations

- Use CI to automate builds and testing of code every time a team member commits changes to version control.
- Include unit tests for IaC and application code as part of your build process.
- If possible, use Microsoft-hosted pool rather than self-hosted pools, as they offer isolation and a clean VM for each pipeline run.
- When you connect Azure DevOps or GitHub to Azure via service connections or GitHub secrets, make sure you always define the scope so they can access only required resources.
- Use Key Vault secrets to avoid hard-coding sensitive information such as credentials (virtual machine's user passwords), certificates, or keys. Then use secrets as variables in your build and release jobs.

Deployment strategy

Design considerations

- Consider using [Continuous Delivery \(CD\)](#). CD involves building, testing, configuring, and deploying from a build to an environment.

- Consider using [environments](#). Environments allow you to target a collection of resources from a delivery job. Examples of common environment names include:
 - Dev
 - Test
 - QA
 - Staging
 - Production
- Consider using IaC as part of your strategy to validate and confirm changes predeployment.
 - [ARM Templates what-if](#)
 - [Bicep what-if](#)
 - [Terraform plan ↗](#)

Design recommendations

- Use CD to ensure that code is always ready to deploy by automatically building, testing, and deploying code to production-like environments. Add continuous delivery to create a full CI/CD integration that helps you detect code defects as early as possible and ensures you can quickly release properly tested updates.
- Use environments as part of your deployment strategy. Environments provide benefits like:
 - Deployment history
 - Traceability of commits and work items
 - Diagnostic resource health
 - Security
- Include IaC predeployment checks so you can preview changes and see details on whether a resource is created, modified, or deleted.

Rollback strategy

Design considerations

- Consider creating a rollback plan. Rolling back a deployment involves reverting the deployment to a known good state and provides a crucial ability to recover from a failed deployment.
- Consider using [undo changes](#) in Git if you need to revert changes in a commit, discard changes, or reset a branch to a previous state.

Design recommendations

- Adopt the use of undo changes in Git when you need to revert changes to committed files, discard uncommitted changes, or reset a branch to a previous state.
-

Feedback

Was this page helpful?

 Yes

 No

Infrastructure as Code

Article • 05/22/2023

Infrastructure as Code (IaC) is a key DevOps practice that involves the management of infrastructure, such as networks, compute services, databases, storages, and connection topology, in a descriptive model. IaC allows teams to develop and release changes faster and with greater confidence. Benefits of IaC include:

- Increased confidence in deployments
- Ability to manage multiple environments
- Improved understanding of the state of infrastructure

For more information about the benefits of using Infrastructure as Code, see [Repeatable infrastructure](#).

Tooling

There are two approaches you can take when implementing Infrastructure as Code.

- **Imperative Infrastructure as Code** involves writing scripts in languages like Bash or PowerShell. You explicitly state commands that are executed to produce a desired outcome. When you use imperative deployments, it's up to you to manage the sequence of dependencies, error control, and resource updates.
- **Declarative Infrastructure as Code** involves writing a definition that defines how you want your environment to look. In this definition, you specify a desired outcome rather than how you want it to be accomplished. The tooling figures out how to make the outcome happen by inspecting your current state, comparing it to your target state, and then applying the differences.

ARM Templates

Review information about Azure Resource Manager templates (ARM templates).

- [What are ARM templates?](#)
- [ARM Templates - Infrastructure as Code overview](#)

Bicep

[Bicep](#) is a domain-specific language (DSL) that uses declarative syntax to deploy Azure resources. In Bicep files, you define the infrastructure you intend to deploy and its

properties. Compared to ARM templates, Bicep files are easier to read and write for a non-developer audience because they use a concise syntax.

Bicep

```
param location string = resourceGroup().location
param storageAccountName string =
'toyleaunch${uniqueString(resourceGroup().id)}'
resource storageAccount 'Microsoft.Storage/storageAccounts@2021-06-01' = {
  name: storageAccountName
  location: location
  sku: {
    name: 'Standard_LRS'
  }
  kind: 'StorageV2'
  properties: {
    accessTier: 'Hot'
  }
}
```

Terraform

Review information about Terraform.

- [Overview of Terraform on Azure](#)
- [Terraform - Infrastructure as Code overview](#)

Azure CLI

Review information about Azure CLI.

- [Azure Command-Line Interface \(CLI\)](#)
- [Azure CLI - Infrastructure as Code overview](#)

Infrastructure as Code modules

One of the goals of using code to deploy infrastructure is to avoid duplicating work or creating multiple templates for the same or similar purposes. Infrastructure modules should be reusable and flexible and should have a clear purpose.

Modules are independent files, typically containing set of resources meant to be deployed together. Modules allow you to break complex templates into smaller, more

manageable sets of code. You can ensure that each module focuses on a specific task and that all modules are reusable for multiple deployments and workloads.

Bicep modules

Bicep allows you to create and call modules. Once modules are created, they can be consumed from any other Bicep template. A high quality Bicep module should define multiple related resources. For example, when you define an Azure function, you typically deploy a particular application, a hosting plan for that application, and a storage account for that application's metadata. These components are separately defined, but they form a logical grouping of resources, so you should consider defining them together as a module.

Bicep modules commonly use:

- **Parameters** to accept values from a calling module.
- **Output values** to return results to a calling module.
- **Resources** to define one or more infrastructure objects for a module to manage.

Publish Bicep modules

You have several options for publishing and sharing Bicep modules.

- **Public registry:** The public module registry is hosted in a Microsoft container registry (MCR). Its source code and the modules it contains are stored in [GitHub](#).
- **Private registry:** You can use Azure container registry to publish modules to a private registry. For information on publishing modules to a registry in a CI/CD pipeline, see [Bicep and GitHub Actions](#), or if you prefer, [Bicep and Azure Pipelines](#).
- **Template Spec:** You can use [template specs](#) to publish Bicep modules. Template specs are meant to be complete templates, but Bicep allows you to use template specs to deploy modules.
- **Version control system:** You can load modules directly from version control tools like GitHub or Azure DevOps.

Terraform modules

Terraform allows you to create and call modules. Each Terraform configuration has at least one module, known as its *root module*, consisting of resources defined in `.tf` files in your main working directory. Each module can call other modules, which allows you to include child modules in your main configuration file. Modules can also be called multiple times within the same configuration or from different configurations.

Modules are defined with all of the same configuration language concepts. They most commonly use:

- **Input variables** to accept values from a calling module.
- **Output values** to return results to a calling module.
- **Resources** to define one or more infrastructure objects for a module to manage.

Publishing Terraform modules

You have several options for publishing and sharing Terraform modules:

- **Public registry:** HashiCorp has their own Terraform Module Registry that allows users to generate sharable Terraform modules. There are currently several [Azure modules](#) published in the Terraform Module Registry.
- **Private registry:** You can seamlessly publish Terraform modules to a private repository like Terraform Cloud Private Registry or Azure Container Registry.
- **Version control system:** You can load private modules directly from version control tools like GitHub. For information on supported sources, see [Terraform module sources](#).

Design considerations

- Consider using IaC when deploying landing zone resources to Azure. IaC fully realizes deployment optimization, reduces configuration effort, and automates the entire environment's deployments.
- Determine whether you should take an imperative or declarative IaC approach.
 - If taking an imperative approach, explicitly state commands to be executed that produce your desired outcome.
 - If taking a declarative approach, specify your desired outcome rather than how you want it done.
- Consider deployment scopes. Have a good understanding of [Azure management levels and hierarchy](#). Each IaC deployment must know the scope at which Azure resources are deployed.
- Determine whether you should use an Azure native or Azure non-native IaC tool. Some points to consider:
 - Azure native tools like Azure CLI, ARM Templates, and Bicep are fully supported by Microsoft, which allows their new features to be integrated faster.

- Non-native tools like Terraform allow you to manage infrastructure as code across multiple cloud providers like AWS or GCP. However, new Azure features can take some time to be included in non-native. If your organization is multicloud or your organization is already using and well-versed in Terraform, consider using Terraform to deploy Azure landing zones.
- Since modules enable you to break complex templates into smaller sets of code, consider using IaC modules for resources that are commonly deployed together. You can ensure each module focuses on a specific task and is reusable for multiple deployments and workloads.
- Consider adopting a publishing strategy for IaC modules by choosing between public registries, private registries or a version control system like a Git repository.
- Consider using a CI/CD pipeline for IaC deployments. A pipeline enforces the reusable process you set to ensure the quality of your deployments and Azure environment.

Design recommendations

- Adopt an IaC approach to deploying, managing, governing, and supporting Azure landing zone deployments.
- Use Azure native tools for IaC in the following scenarios:
 - You want to use only Azure native tools. Your organization might have prior ARM or Bicep template deployment experience.
 - Your organization wants to have immediate support for all preview and GA versions of Azure services.
- Use non-native tools for IaC in the following scenarios:
 - Your organization currently uses Terraform to deploy infrastructure to other clouds like AWS or GCP.
 - Your organization doesn't need to have immediate support for all preview and GA versions of Azure services.
- Use reusable IaC modules to avoid repetitive work. You can share modules across your organization to deploy multiple projects or workloads and manage less complex code.
- Publish and use IaC modules from public registries in the following scenarios:

- You want to use modules for Azure Landing Zone already published to public registries. For more information, see [Azure landing zones Terraform module](#).
 - You want to use modules that are maintained, updated, and supported by Microsoft, Terraform, or other module providers.
 - Make sure you check the support statement from any module provider you evaluate.
- Publish and use IaC modules from private registries or version control systems in the following scenarios:
 - You want to create your own modules based on your organizational requirements.
 - You want to have full control of all features and maintain, update, and publish new versions of modules.
- Use a CI/CD pipeline to deploy IaC artifacts and ensure the quality of your deployment and Azure environments.

Environments

Article • 01/25/2023

Use the [Continuous Delivery](#) process to quickly and safely delivers new value to production. You can deliver small changes frequently, which reduces the risk of problems.

Other factors affect "deployment pain to production", including your adoption of multiple delivery/deployment environments. A multienvironment approach lets you build, test, and release code with greater speed and frequency to make your deployment as straightforward as possible. You can remove manual overhead and the risk of a manual release, and instead automate development with a multistage process targeting different environments.

A common multienvironment architecture includes four tiers:

- Development
- Test
- Staging
- Production

In this architecture, your product transitions in order from Development (the environment where you develop changes to the software) through Production (the environment your users directly interact with). You might also introduce a User Acceptance Test (UAT) environment to validate end-to-end business flow.

Environment	Description
Development	Your development environment (dev) is where changes to software are developed.
Test	Your test environment allows either human testers or automated tests to try out new and updated code. Developers must accept new code and configurations through unit testing in your development environment before allowing those items to enter one or more test environments.
Staging	Staging is where you do final testing immediately prior to deploying to production. Each staging environment should mirror an actual production environment as accurately as possible.
UAT	User Acceptance Testing (UAT) allows your end-users or clients to perform tests to verify/accept the software system before a software application can move to your production environment.

Environment	Description
Production	Your production environment (production), sometimes called <i>live</i> , is the environment your users directly interact with.

Design considerations

Apply the following considerations to both Azure Landing Zones and Azure Workloads development:

- Test environments are important because they allow platform developers to test changes before deploying to production, which reduces risk related to delivery in production.
- Keeping your environments as similar as possible makes it easy to find environment-related errors in the first phases of testing, which increases development and testing speed and reliability.
- If there are discrepancies in the configuration of your environments, "configuration drift" happens, which can result in data loss, slower deployments, and failures.
- You can speed up deployments, improve environment consistency, and reduce "configuration drift" between environments by adopting Infrastructure as Code (IaC).
- Consider adopting methods like Canary or Blue-Green Deployments that make new features available only to a limited set of test users in production and help reduce the time to release into production.
- Use checks on test results to control the transition of code from development to production. You can automate these controls so that failing tests prevents changes from automatically deploying to the next environment.
- Have designated users review pull requests before code is deployed to production. Consider using repositories with [branch strategy](#) to manage the review process.
- Avoid silos by allowing all developers to access all environments.

Workloads

To learn how to manage environments for Workloads refer to [Enterprise-scale FAQ](#).

Azure Landing Zones

Adopting multiple environments for an Azure Landing Zone deployment is common when a customer wants to test the effects and results of new Azure Policy Assignments,

Azure RBAC role assignments, Azure AD group memberships, Azure resources' creation, and more.

[Testing approach for enterprise-scale](#) describes two different adoption approaches:

- Replication of management group hierarchy in Canary and Production environment
- Sandbox subscriptions

Regardless of which approach you follow, you should always:

- Adopt at least one environment for testing.
- Use separated Service Principals for test and production purposes to protect your environments.
- Implement automated checks and approvals to validate and approve changes prior to deploying any change to a particular environment

Next steps

- [Create and target an environment](#)
- [Using environments for deployment \(GitHub\)](#) ↗

Test-driven development for Azure landing zones

Article • 10/19/2022

Test-driven development (TDD) is a software development and DevOps process that improves the quality of new features and improvements in code-based solutions. TDD creates unit test cases before developing the actual code, and tests the code against the test cases. This approach is opposed to developing code first and creating test cases later.

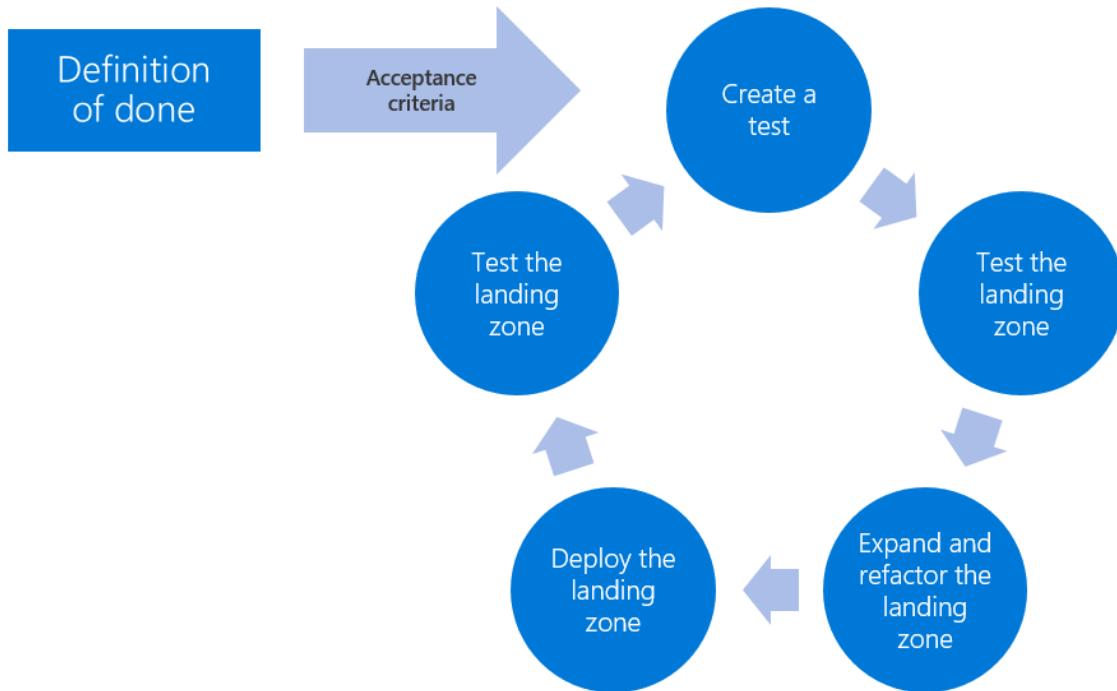
A [landing zone](#) is an environment for hosting workloads that is preprovisioned through code. Landing zones include foundational capabilities that use a defined set of cloud services and best practices. This article describes an approach that uses TDD to deploy successful landing zones while meeting quality, security, operations, and governance requirements.

Cloud infrastructure is the output of code execution. Well-structured, tested, and verified code produces a viable landing zone. Cloud-based infrastructure and its underlying source code can use this approach to ensure that landing zones are high quality and meet core requirements.

Use this approach to meet simple feature requests during early development. Later in the cloud adoption lifecycle, you can use this process to meet security, operations, governance, or compliance requirements. The process is especially useful for developing and refactoring landing zones in a parallel development effort.

Test-driven development cycle

The following diagram shows the test-driven development cycle for Azure landing zones:



1. **Create a test.** Define a test to validate that acceptance criteria for a feature has been met. Automate the test as you develop, to reduce the amount of manual test effort, especially for enterprise-scale deployments.
2. **Test the landing zone.** Run the new test and any existing tests. If the required feature isn't included in the cloud provider's offerings and hasn't been provided by prior development efforts, the test should fail. Running existing tests helps validate that your new feature or test doesn't reduce the reliability of existing landing zone features.
3. **Expand and refactor the landing zone.** Add or modify source code to fulfill the requested value-add feature and improve the general quality of the code base.

To meet the TDD criteria, the cloud platform team would add code only to meet the requested feature. However, code quality and maintenance are shared efforts. As they fulfill new feature requests, the cloud platform team should try to improve code by removing duplication and clarifying the code. Running tests between new code creation and refactoring of source code is highly recommended.
4. **Deploy the landing zone.** Once the source code fulfills the feature request, deploy the modified landing zone to the cloud provider in a controlled testing or sandbox environment.
5. **Test the landing zone.** Retest the landing zone to validate that the new code meets the acceptance criteria for the requested feature. Once all tests pass, the feature is considered complete and the acceptance criteria are considered met.

The TDD cycle repeats the preceding basic steps until they meet the full *definition of done*. When all value-added features and acceptance criteria pass their associated tests, the landing zone is ready to support the next wave of the cloud adoption plan.

The cycle that makes TDD effective is often referred to as a *red/green test*. In this approach, the cloud platform team starts with a failed test, or red test, based on the definition of done and the defined acceptance criteria. For each feature or acceptance criteria, the cloud platform team completes development tasks until the test passes, or has a green test.

The goal of TDD is to address better design, not to create a suite of tests. The tests are a valuable artifact for completing the process.

Definition of done

Success can be a subjective measure that provides a cloud platform team little actionable information during landing zone development or refactoring. Lack of clarity can lead to missed expectations and vulnerabilities in a cloud environment. Before the cloud platform team refactors or expands any landing zones, they should seek clarity regarding the *definition of done* (DoD) for each landing zone.

DoD is a simple agreement between the cloud platform team and other affected teams that defines the expected value-added features to include in the landing zone development effort. The DoD is often a checklist that's aligned with the short-term cloud adoption plan.

As teams adopt more workloads and cloud features, the DoD and the acceptance criteria become more complex. In mature processes, the expected features each have their own acceptance criteria to provide more clarity. When the value-added features all meet the acceptance criteria, the landing zone is sufficiently configured to enable the success of the current adoption wave or release.

Simple DoD example

For an initial migration effort, the DoD might be overly simple. The following example is a simple DoD:

The initial landing zone will host 10 workloads for initial learning purposes. These workloads aren't critical to the business and have no access to sensitive data. In the future, these workloads will probably release to production, but the criticality and sensitivity aren't expected to change.

To support these workloads, the cloud adoption team needs to meet the following criteria:

- Network segmentation to align with proposed network design. This environment should be a perimeter network with access to the public internet.
- Access to compute, storage, and networking resources to host the workloads aligned to the digital estate discovery.
- Naming and tagging schema for ease of use.
- During adoption, temporary access for the cloud adoption team to change service configurations.
- Prior to production release, integration with the corporate identity provider to govern ongoing identity and access for operations management. At that time, the cloud adoption team's access should be revoked.

The last point isn't a feature or acceptance criterion, but an indicator that more expansions will be required and should be explored with other teams early.

More complex DoD examples

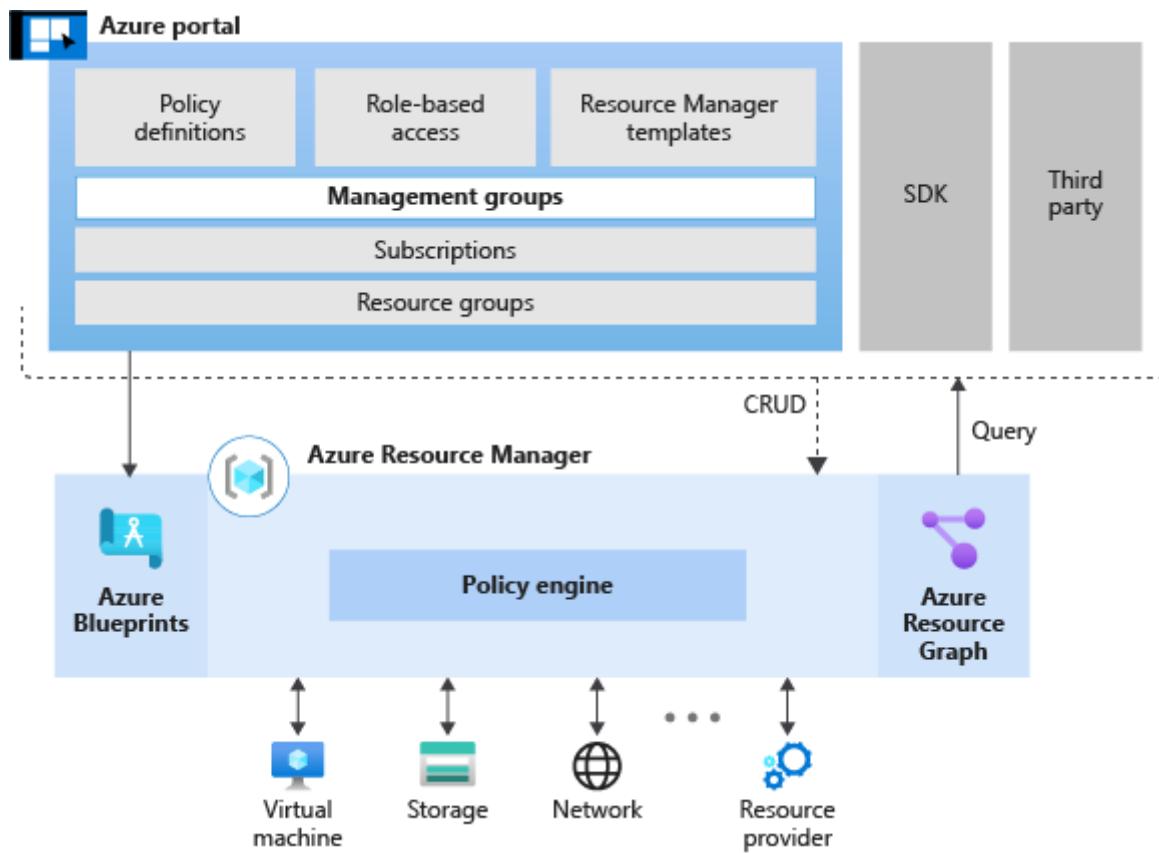
The Govern methodology within the Cloud Adoption Framework provides a narrative journey through the natural maturity of a governance team. Embedded in that journey are several examples of DoD and acceptance criteria, in the form of policy statements.

- **Initial policy statements.** Example of initial DoD based on corporate policies governing early stage adoption requirements.
- **Incremental improvements to expand identity management.** Example of corporate policies governing DoD to meet requirements to expand identity management for a landing zone.
- **Incremental improvements to expand security requirements.** Example of corporate policies governing DoD to meet security requirements aligned to the reference cloud adoption plan.
- **Incremental improvements to expand operations management.** Example of corporate policies governing DoD to meet basic operations management requirements.
- **Incremental improvements to expand cost management.** Example of corporate policies governing DoD to meet cost management requirements.

The preceding examples are basic samples to help you develop a DoD for your landing zones. You can get sample policies for each of the [Five Disciplines of Cloud Governance](#).

Azure tools and features to support landing zone TDD

The following diagram shows available test-driven development tools in Azure:



You can easily integrate these Azure tools and features into TDD for landing zone creation. The tools serve specific purposes, making it easier to develop, test, and deploy landing zones in alignment with TDD cycles.

- **Azure Resource Manager** provides a consistent platform for build and deployment processes. The Resource Manager platform can deploy landing zones based on source code definitions.
- **Azure Resource Manager (ARM) templates** provide primary source code for environments deployed in Azure. Some third-party tools like Terraform provide their own ARM templates to submit to Azure Resource Manager.
- **Azure Quickstart Templates** provide source code templates that help accelerate landing zone and workload deployment.
- **Azure Policy** provides the primary mechanism for testing acceptance criteria in your DoD. Azure Policy can also provide automated detection, protection, and resolution when deployments deviate from governance policies.

In a TDD cycle, you can create a policy definition to test a single acceptance criteria. Azure Policy includes [built-in policy definitions](#) that can meet individual acceptance criteria within a DoD. This approach provides a mechanism for red tests before you modify the landing zone.

Azure Policy also includes [built-in policy initiatives](#) that you can use to test and enforce the full DoD for a landing zone. You can add all acceptance criteria to a policy initiative assigned to the entire subscription. Once the landing zone meets the DoD, Azure Policy can enforce the test criteria to avoid code changes that would cause the test to fail in future releases.

Design and review [Azure Policy as Code workflows](#) as part of your TDD approach.

- [Azure Blueprints](#) groups policies and other deployment tools into a repeatable package that you can assign to multiple landing zones. Blueprints are useful for multiple adoption efforts that share common DoDs, which you might want to update over time. Azure Blueprints can also help with deployment during subsequent efforts to expand and refactor landing zones.

Azure Blueprints provides various [blueprint samples](#), including policies for testing and templates for deployment. These blueprint samples can accelerate development, deployment, and testing efforts in TDD cycles.

- [Azure Resource Graph](#) provides a query language for creating data-driven tests based on information about the assets deployed in a landing zone. Later in the adoption plan, this tool can also define complex tests based on the interactions between workload assets and the underlying cloud environment.

Resource Graph includes advanced [query samples](#), which you can use to understand how workloads are deployed within a landing zone for advanced testing scenarios.

Depending on your preferred approach, you can also use the following tools:

- [Deploy landing zones using Terraform](#).
- [Deploy landing zones using Bicep ↗](#).
- [Manage landing zones using AzOps ↗](#), a PowerShell module that pushes resource templates and Bicep files at all Azure scope levels, and pulls and exports Azure resource hierarchies.

Next steps

- [Security considerations for DevOps platforms](#)

- Landing zone implementation options

DevOps toolchain

Article • 01/04/2023

A DevOps toolchain is a collection of tools that enables DevOps teams to collaborate across the entire product lifecycle and to tackle key DevOps fundamentals.

The tools a DevOps toolchain includes operate as an integrated unit for planning, continuous integration, continuous delivery, operations, collaboration and feedback. You can review some examples of DevOps technologies across different DevOps stages in [Define your DevOps technology ecosystem](#).

DevOps toolchain considerations

- DevOps' processes can already be in use across your organization when you select a toolchain. You should find the right balance between the adoption of technologies suitable for your team's needs and the goal of standardizing and avoiding heterogeneous DevOps ecosystems across your organization.
- You can adopt different kinds of DevOps toolchains:
 - **All-in-one:** Provides a complete solution that might not integrate with other third-party tools. All-in-one toolchains can be useful for organizations beginning their DevOps journey. Example: [Full stack Azure DevOps toolchain](#).
 - **Customized:** Allows teams to bring and mix existing tools they know and already have in use into the wider DevOps toolchain. Integration is essential for these types of toolchains to avoid spending unnecessary time switching between screens, logging in to multiple places, and having the challenge to share information between tools. Example: [Azure DevOps and GitHub toolchain](#).
- Consider using toolchains that are regularly updated and that have assistance available whenever you need it through email or online portal. This is a requirement for any product or service that is on the critical path to market.

Planning

- Consider adopting a tool that supports [Continuous Planning](#) practices:
 - Release planning
 - Epic and feature identification
 - Prioritization
 - Estimation
 - User story definition

- Backlog refinement
- Sprint planning
- Daily Scrum
- Sprint review
- Retrospective

Continuous Integration and Continuous Delivery

- When implementing [Continuous Integration \(CI\)/Continuous Delivery \(CD\)](#), consider adopting a tool that supports:
 - Version Control Systems. Everything in your project must be checked in to a single version control repository like Git: code, tests, database scripts, build and deployment scripts, and anything else needed to create, install, run and test your application.
 - [Branching strategy](#).
 - [Automated builds](#).
- Note that your choice of repository is also influenced by data sovereignty/residency requirements. If you need your data to be hosted locally in country/region other than US, you'll need Azure DevOps repositories when GitHub Repos can't be used.
- To minimize the amount of manual configuration required to provision resources, consider adopting [Infrastructure as Code \(IaC\)](#). IaC lets you apply software engineering practices like testing and versioning, which make infrastructure and deployments automated, consistent, and repeatable. Keep scripts and templates under source control like any other code you maintain.
- Adopt [code scanning tools](#) to help you detect code defects as soon as possible. Include pre-deployment checks to validate and confirm changes before any deployment (Example: "[what-if](#)") function.
- CI/CD tools speed up the time to market for your product. Tools that allow you to parallelize tasks and take advantage of elastic scalability on cloud-hosted infrastructure enhance the performance of your CI/CD process.
- Consider using CI/CD tool features that support the measure of DevOps performance. Dashboards and reporting can track aspects of your development process like lead time, cycle time, velocity of work, and so on.

Continuous Operations

Continuous Operations is a focus that helps organizations maintain continuity of output between internal systems and customers through the uninterrupted delivery of critical services or functions. The goals of Continuous Operations are:

- To reduce or eliminate the need for planned downtimes or interruptions such as scheduled maintenance, capacity optimization and deployment.
- To increase overall reliability and resiliency of systems in three aspects, with people, process and tools.

Use cloud-native tools to:

- Monitor key metrics for service performance and availability.
- Gain digital experience and customer insights.
- [Generate intelligence-driven responses](#) for incidents, system recovery, or scaling.
 - [Azure Diagnostics](#) and [Application Insights](#) are the standard method for tracking the health and status of Azure resources. [Azure Monitor](#) also provides centralized monitoring and management for cloud or hybrid solutions.
- Automate proactive maintenance and tasks like deployment or system updates.
 - [Azure Automation](#) is a cloud-native tool you can use to create event-based automation to diagnose and resolve issues.

Collaboration and feedback

- Rapid feedback loops are at the heart of the CI/CD process. A CI/CD tool uses feedback to resolve conditions in CI/CD workflow logic and displays information back to users, usually through a dashboard.
- Support for email notifications and integration with IDEs or communication platforms ensure you can stay informed about what's happening without having to check a dashboard. Ensure you have the flexibility to configure which alerts you receive, since getting too many alerts transforms them into background noise.
- Any tool you choose for the collaboration should support the following collaboration practices:
 - Kanban collaboration
 - Wiki content collaboration
 - ChatOps collaboration
 - Team room

DevOps toolchain recommendations for Azure Landing Zones

DevOps toolchains for Azure Landing Zone implementation should consider all previously discussed DevOps phases:

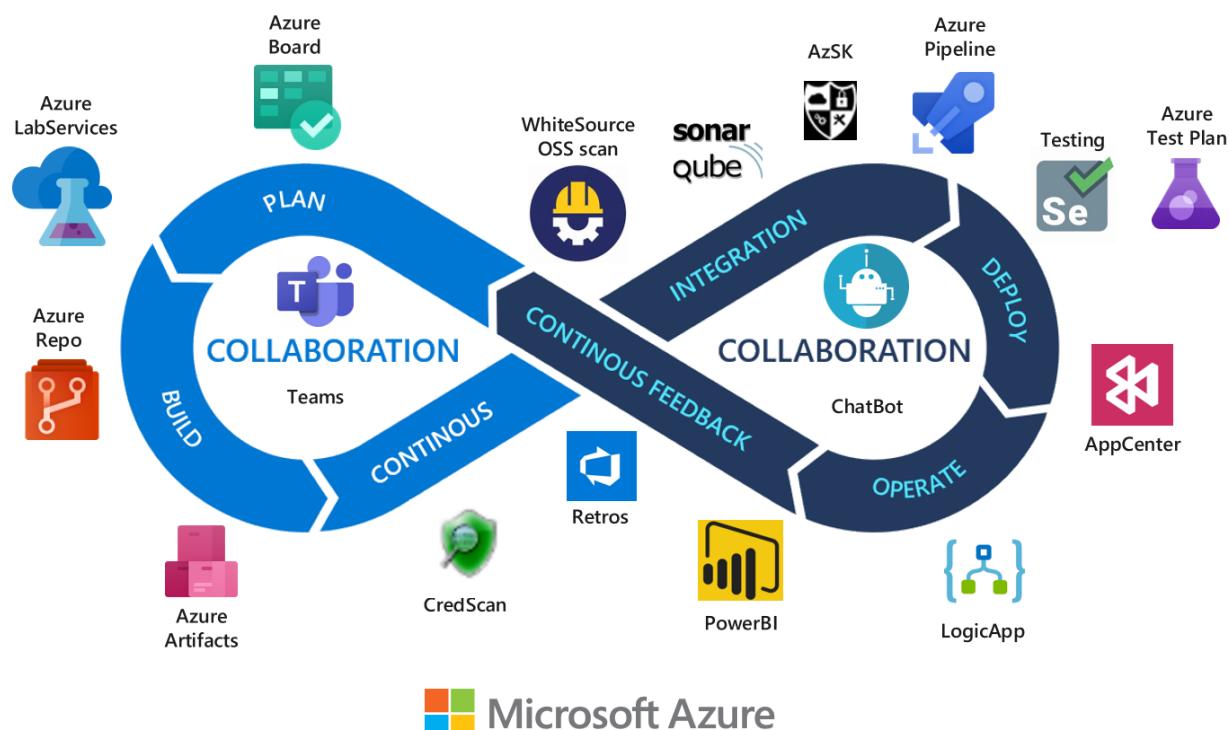
- Planning
- CI/CD (including automation capabilities like Infrastructure as code)
- Operations
- Collaborations and feedback

Review guidance for landing zone deployment and considerations for choosing an implementation option in [Choosing landing zone adoption](#).

Regardless of selected methodology (start small and expand or enterprise-scale), there are a few common topologies that enterprises tend to follow as they design their DevOps workflows and toolchains.

- **Full stack Azure DevOps toolchain:** For enterprises that are already heavily invested in the Microsoft ecosystem, this topology allows them to take full advantage of the native integrations between Microsoft products and services and streamline key processes.
- **Azure DevOps and GitHub toolchain:** This topology allows you to use the strengths of both Azure and GitHub as part of a well-integrated solution.

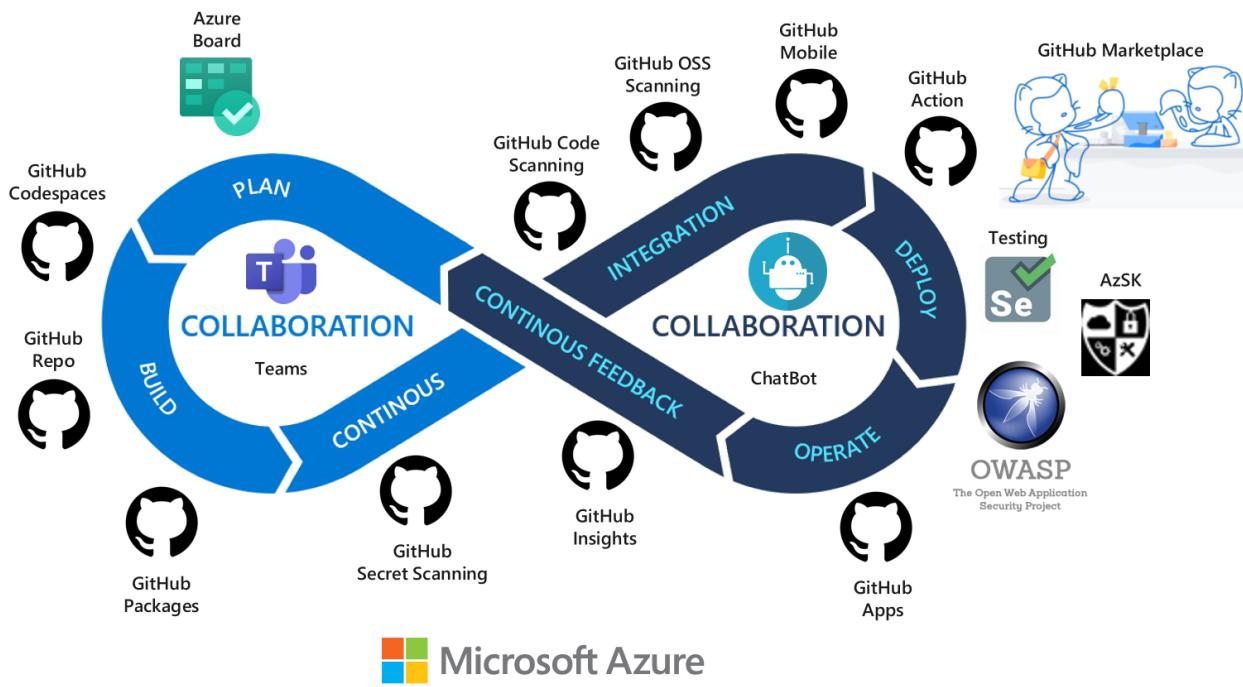
Full stack Azure DevOps toolchain



DevOps stage	tools
Planning	Azure Boards provides powerful and flexible planning capabilities to developers and other personas, including hierarchical backlogs, customizable Kanban boards, rich process customization, team dashboards, and custom reporting.

DevOps stage	tools
CI/ CD	<p>Azure Repos allows you to create private Git repositories, and it supports different Git clients, branching strategy, and protection. Azure Repos also provides localized data residency in the cloud to enable compliance with European regulations. Azure Pipelines allows customers to set up automated pipelines for CI/CD, including for advanced test reporting, and provides powerful support for multistage pipelines. Fine-grained permissions, gates, custom checks, and automated test result reporting in Azure Pipelines help you enforce security, compliance, and safe deployment best practices in your organization and support parallel steps execution and scalability. Azure Artifacts provides a feed to store packages and to review and validate each package for security purposes, and also provides granular permission control and auditing. Azure Test Plans in Azure DevOps provides a browser-based test management solution for exploratory, manual, and user acceptance testing. Users of Azure Test Plans also typically use Azure Boards for planning and project management. You can link user stories and other requirements to test cases and can document bugs found through testing. Adopt Marketplace extensions for DevOps to improve static code analysis with tools like Credential scanners, open-source scanners, Bugs and Vulnerabilities scanners, and more.</p>
Operations	<p>Azure Dashboards and reporting provide custom reporting to help you monitor key service performance metrics. Azure Diagnostics and Application Insights are the standard method of tracking the health and status of Azure resources. Azure Monitor provides centralized monitoring and management. Azure Automation can be used to create event-based automation to diagnose and resolve issues.</p>
Collaborations and Feedbacks	<p>Azure DevOps Wiki allows you to share information with members of other teams and supports collaborative editing of its content and structure. Azure Boards provide Kanban collaboration and support for comments and discussions in backlog items. You can integrate Microsoft Teams with Azure DevOps for a complete team collaboration experience.</p>

Azure DevOps and GitHub toolchain



DevOps stage	tools
Planning	Azure Boards provides a stable and scalable solution for planning, repository management, data visualization, and hierarchical work item organization. It integrates with GitHub, so you can link work items and GitHub commits. It also allows you to choose your ideal workflow, whether that's a simple, out-of-the-box workflow or a custom workflow you build with the powerful and flexible Azure Boards customization engine. When you need to visualize your data, Azure Boards helps you easily build and configure custom dashboards and monitor progress throughout your project lifecycles.
CI/ CD	Use the GitHub Enterprise (GHE) version of GitHub, which includes GitHub Repo and GitHub Advanced Security (GHAS). GHAS includes CodeQL, Code Scanning, Secret Scanning, and Dependency Review. GHE also offers Codespaces, a cloud IDE you can use to develop code and that can replace Visual Studio Code, which organizations usually include in full-stack Azure DevOps scenarios. You can use GitHub Actions to automate non-build workflows if your repositories are in GitHub. If you have more complex scenarios where you need to access code from outside of GitHub or require centralized management for workflow templates and build pipelines, adopt Azure Pipelines. For Azure Boards, you can integrate Azure Pipelines with GitHub repositories. To learn about integrating Azure DevOps and GitHub, see Work with Azure DevOps and GitHub. GitHub Packages is a software package service that allows you to host your own packages privately or publicly. GitHub offers container registry support for hosting Docker or OCI images. You need access tokens in order to publish, install, or delete packages and keep your package lifecycle management secure. To automate packages, you can integrate GitHub Packages with GitHub Actions, GitHub APIs, and webhooks to create DevOps workflows that include code, CI, and deployments all in one interface.

DevOps stage	tools
Operations	<p>GitHub Insights provides analytic reports based on data from your GitHub Enterprise Server instance to help you understand and improve your software delivery process. For Landing Zone diagnostics and management, use the Azure services recommended in the Full-stack Azure DevOps scenario.</p>
Collaboration and feedback	<p>You can use GitHub Discussions to share questions, ideas, conversations, requests for comment (RFC), resource planning, and announcements. Use Azure Boards to easily build and configure custom dashboards and monitor progress throughout your project lifecycles. Adopt Microsoft Teams for a complete team collaboration experience.</p>

Security considerations for DevOps platforms

Article • 10/09/2023

Security should always be a priority in cloud-based development platforms such as [Azure DevOps](#) and [GitHub](#). Microsoft updates and maintains the security of the underlying cloud infrastructure, but it's up to you to review and configure security best practices for your own Azure DevOps organizations and GitHub instances.

Consider the following critical security areas whether you deploy environments through infrastructure as code in continuous integration and continuous deployment (CI/CD) pipelines, or deploy code to your applications hosted in Azure.

Restrict access to DevOps tooling

Follow the principle of least privilege by using role-based access control (RBAC) through [Microsoft Entra ID](#). Give users and services the minimum amount of access to your DevOps platforms that they need to do their business functions. For more information, see the following articles:

- [Connect your organization to Microsoft Entra ID](#)
- [Microsoft Entra Single Sign-On \(SSO\) integration with GitHub Enterprise Cloud](#)
- [Azure DevOps security best practices](#)

After you establish Microsoft Entra ID as your identity management plane, follow best practices to manage Azure DevOps role assignments with [Microsoft Entra group memberships](#). You can [assign Azure DevOps roles to Microsoft Entra groups](#), and adjust a user's Microsoft Entra membership to change or remove their Azure DevOps access.

- Use Microsoft Entra ID [entitlement management](#) to create access packages that allow Microsoft Entra users time-bound access to required resources to complete their tasks.
- You can also use Microsoft Entra [Privileged Identity Management](#) for just-in-time access to promote individuals to Azure DevOps Administrator roles for a period of time.

Manage security in Azure DevOps by using security groups, policies, and settings at the Azure DevOps organization, project, or object level. Consider disabling permission inheritance in Azure DevOps if possible.

Restrict repository and branch access

Restrict repository access, permissions, and branch creation to protect your code and environments from undesired or malicious changes. [Restrict access to repositories](#) by using security groups in Azure DevOps. Limit who can read and update code in your branches by setting [branch permissions](#).

Restrict pipeline access and permissions

Malicious code might steal enterprise data and secrets, and corrupt production environments. Implement guardrails to prevent malicious code deployment in the pipeline. By restricting access and implementing guardrails, you can also prevent lateral exposure to other projects, pipelines, and repositories from any compromised pipelines.

Consider following an incremental approach to securing your YAML pipelines. For more information, see [Plan how to secure your YAML pipelines](#).

Select the DevOps agent based on security needs

You can use Microsoft-hosted or self-hosted agents to power Azure DevOps and GitHub pipelines. There are trade-offs for each type of agent.

With Microsoft-hosted agents, you don't need to worry about upgrades or maintenance. With self-hosted agents, you have greater flexibility to implement security guardrails. You control the agent hardware, operating system, and installed tools.

See [Azure Pipelines agents](#) to review the differences between the types of agents and identify potential security considerations.

Use secure and scoped service connections

Whenever possible, use a [service connection](#) to deploy infrastructure or application code in an Azure environment. The service connection should have limited deployment access to specific Azure resources or resource groups, to reduce any potential attack surfaces. Also, consider creating separate service connections for development, testing, QA, and production environments.

Use a secret store

Never hard-code secrets in code or auxiliary documentation in your repositories. Adversaries scan repositories, searching for exposed confidential data to exploit. Set up a secret store such as [Azure Key Vault](#), and reference the store in Azure Pipelines to securely retrieve keys, secrets, or certificates. For more information, see [Secure the pipeline and CI/CD workflow](#). You can also [use Key Vault secrets in GitHub Actions workflows](#).

Use hardened DevOps workstations to build and deploy code

Platform and development teams often have elevated privileges on the Azure platform, or on other services such as Azure DevOps and GitHub. This access greatly increases the potential attack surface. Implement guardrails to secure any endpoints and workstations you use to develop and deploy code.

Use hardened [secure admin workstations \(SAWs\)](#) to deploy any changes to high-risk and production environments. For more information, see [Secure endpoints with Zero Trust](#).

Do security scanning and testing

Whether you deploy application code or infrastructure as code, implement [DevSecOps best practices and controls](#) in your pipelines. Integrate security early in your CI/CD journey to prevent costly security breaches later on. Create a strategy to implement static code analysis, unit testing, secret scanning, and package/dependency scanning in your pipelines.

Enterprise security tools such as [Microsoft Defender for Cloud](#) can integrate with DevOps tools. For example, Defender for Cloud can [identify vulnerable container images in your CI/CD workflows](#). For GitHub Actions and repositories, use [GitHub Advanced Security](#) for code and secret scanning and dependency review.

Periodically review audit events to monitor and react to unexpected usage patterns by administrators and other users. You can [access, filter, and export audit logs](#) for your Azure DevOps organization. For long-term storage and detailed log querying, [create an audit stream](#) to an [Azure Monitor Log Analytics](#) workspace, or to a security information and event management (SIEM) system like [Microsoft Sentinel](#).

Role-based access control for DevOps tools

Article • 10/09/2023

When you deploy cloud-based solutions for your infrastructure deployments, security should always be your most important concern. Microsoft keeps the underlying cloud infrastructure secure. You configure security in Azure DevOps or GitHub.

Prerequisites

Once you decide which Azure Landing Zone templates to deploy, clone them into your own repository. Set up the CI/CD pipelines. For both GitHub and Azure DevOps, there are several authentication methods available, such as personal access tokens (PAT) and integrating with an identity provider, such as Microsoft Entra ID. For more information, see [Use personal access tokens](#).

We recommend that you integrate with Microsoft Entra ID to use all its capabilities. Integration helps streamline your role assignment process and identity lifecycle management. For more information, see [Connect your organization to Microsoft Entra ID](#). If you're using GitHub, consider [integrating GitHub Enterprise with Microsoft Entra ID](#).

General Design Considerations

We recommended that you maintain tight control of administrators and service account groups across Microsoft Entra ID and your DevOps tool. Consider implementing the principle of least privilege across all your role assignments.

For example, your organization might have a Platform or Cloud Excellence team that maintains Azure Resource Manager templates for your Azure Landing Zones. Assign users on that team to a Security Group in Microsoft Entra ID, assuming that you're using it as your identity provider. Assign roles to that security group in your DevOps tool so those users can do their jobs.

For any administrator or highly privileged accounts in Active Directory, we recommend that the credentials aren't synchronized to Microsoft Entra ID, and vice-versa. This approach reduces the threat of lateral movement. If an administrator in Microsoft Entra ID is compromised, the attacker won't be able to easily gain access to any cloud assets, such as Azure DevOps. That account can't potentially inject malicious tasks in the CI/CD

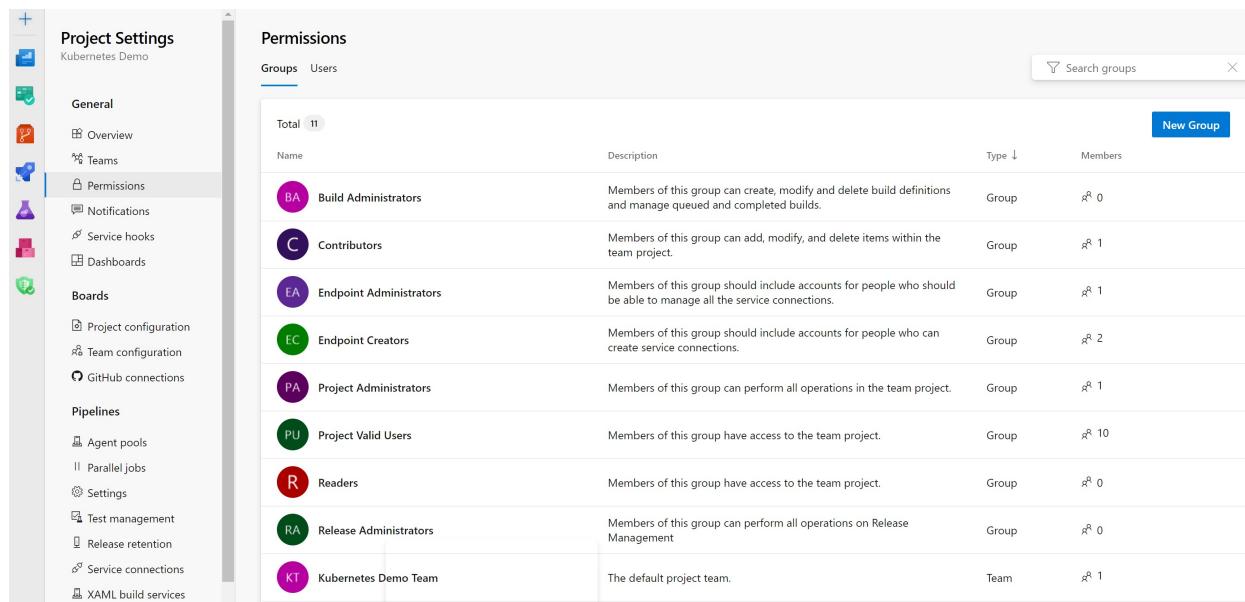
pipelines. This step is particularly important for any users assigned elevated permissions in your DevOps environment, such as Build or Project/Collection Administrators. For more information, see [Security best practices in Microsoft Entra ID](#).

Azure DevOps role-based access considerations

Manage security in Azure DevOps with security groups, policies, and settings at the organization/collection, project, or object level. To integrate with an identity provider such as Microsoft Entra ID, consider creating [conditional access policies to enforce multifactor authentication for all users](#). The policies allow access to your Azure DevOps organization and more granular restrictions around IP address, type of device used for access, and device compliance.

For most team members in your Platform team that manage your Azure Landing Zones, the *Basic* access level and *Contributor* default security group should provide sufficient access. The Contributor security group allows them to edit the Azure Landing Zone templates in your repository and the CI/CD pipelines that validate and deploy them.

We recommend that you assign your Platform team to the Contributor security group at the project level of Azure DevOps. This approach follows the principle of least privilege. These assignments can be done through the **Project Settings** page shown below.



The screenshot shows the 'Permissions' section of the 'Project Settings' page for the 'Kubernetes Demo' project. The left sidebar lists various project settings categories like General, Boards, Pipelines, and Test management. The 'Permissions' section is selected. The main area displays a table of security groups:

Name	Description	Type	Members
Build Administrators	Members of this group can create, modify and delete build definitions and manage queued and completed builds.	Group	0
Contributors	Members of this group can add, modify, and delete items within the team project.	Group	1
Endpoint Administrators	Members of this group should include accounts for people who should be able to manage all the service connections.	Group	1
Endpoint Creators	Members of this group should include accounts for people who can create service connections.	Group	2
Project Administrators	Members of this group can perform all operations in the team project.	Group	1
Project Valid Users	Members of this group have access to the team project.	Group	10
Readers	Members of this group have access to the team project.	Group	0
Release Administrators	Members of this group can perform all operations on Release Management	Group	0
Kubernetes Demo Team	The default project team.	Team	1

Another best practice for your Azure DevOps Projects and organizations is to disable inheritance where possible. Users inherit permissions allowed by their security group assignments. Due to the allow-by-default nature of inheritance, unexpected users can get access or permissions.

For example, if you assign your Platform team Contributor security group membership, verify their permissions on the Azure Landing Zones repository. You should have branch

policies in place to verify that the security group isn't allowed to bypass those policies during pull requests. Verify this setting under [Project Settings > Repositories](#).

After you've assigned permissions to users, periodically review audit events to monitor and react to unexpected usage patterns by administrators and other users. Start by [creating an audit stream to a Log Analytics workspace](#). If your workspace uses Microsoft Sentinel, create analytics rules to alert you on notable events, such as improper use of permissions.

For more information, see the following resources:

- [Azure DevOps security best practices](#)
- [Azure DevOps groups and permissions](#)
- [Azure DevOps access levels](#)

GitHub Role-based Access Considerations

If your primary DevOps tool is GitHub, you can assign users access to resources by granting them roles at the repository level, team level, or organization level. After you fork the Azure Landing Zones repository and integrate with an identity provider, such as Microsoft Entra ID, consider creating a team in GitHub. Assign that team *write* access to your new Azure Landing Zone repository. For most of your Platform team members, who modify and deploy the Landing Zones, write access should be sufficient. For project managers or Scrum managers on the team, you might need to assign them the *Maintain* role to that repository.

We recommend that you manage all of these role assignments through the integrated identity provider. For example, you can synchronize the Platform team for the Azure Landing Zone repository you've created in GitHub with the corresponding Platform team Security Group in Microsoft Entra ID. Then, as you add or remove members to the Microsoft Entra Security Group, those changes are reflected in your GitHub Enterprise Cloud role assignments.

ⓘ Note

Once you connect a specific GitHub team to an integrated identity provider, you're restricted to managing the team membership through it.

Next steps

For more information around managing roles and teams in GitHub, see these resources:

- GitHub Roles and Scope Levels ↗
- How to manage GitHub permissions after integrating with Identity Provider ↗
- Managing Permissions with GitHub Teams ↗

Landing zone implementation options

Article • 02/14/2024

ⓘ Important

The Azure landing zones **Implementation options** section of the Cloud Adoption Framework is undergoing a freshness update.

As part of this update, we will be revising the table of contents and article content, which will include a combination of refactoring and consolidation of several articles. An update will be posted on this page once the work is completed.

Visit the new ["Deployment options" section of the Azure Architecture Center](#) for the latest Azure landing zone implementation content, including platform and application landing zones.

An [Azure landing zone](#) provides cloud adoption teams with a well-managed environment to run their workloads. Take advantage of the best practices described in [landing zone design areas](#) to build a strong foundation. You can then extend the foundation by implementing processes related to security, governance, and compliance.

Environment development approaches

There are two primary approaches. The choice will depend on how fast your teams can develop the required skills.

- **Start with the Azure landing zone accelerator:** If your business requirements call for a rich initial implementation of landing zones with fully integrated governance, security, and operations from the start. If you need to, you can modify using Infrastructure-as-Code (IaC) to set up and configure an environment per your requirements. For IaC, your organization will require skills in Azure Resource Manager templates and GitHub.
- **Customize:** If it's more important to build your environment to meet specific requirements, or develop internal skills. In this approach, focus on the basic landing zones considerations required to start cloud adoption. All technical and business requirements are considered complete when your environment configuration aligns with Azure landing zone conceptual architecture. You can then focus on enhancing your landing zone.

Important

Of the two approaches, the Azure landing zone accelerator is recommended because it's the quickest way to achieve a scaled-out and mature environment.



Deploy to Azure



Beside the use of the Azure landing zone accelerator, there are use cases where organizations have specific business or technical requirements. For those cases, some customization might be needed.

To address the customization use cases, consider the [implementation options](#) given in this article. The options are intended for users with strong skills in technologies such as Azure Resource Manager, Azure Policy, DevOps tools, and third-party deployment tools. Those technologies are required for a solid foundation on which to build a landing zone.

Caution

The best practices used for customization will ultimately be aligned with the [Azure landing zone](#). However, there's added investment in time and effort which might be justified to fit specific business requirements.

Finally, guidance in the [Govern](#) and [Manage](#) methodologies will build on top of your initial landing zones. The design of any Azure landing zone outline will likely require refactoring over time.

Implementation options

Here are some implementation options for landing zones keeping in mind the development approaches described above. Each implementation option in this table is designed for a specific set of operating model dependencies to support your organizations nonfunctional requirements. Every option includes distinct automation approaches and tools. Even though each option is mapped to a different operating model, they have common design areas. The difference is how you choose to implement them and the level of technical experience required.

Azure landing zone accelerator approach

 Expand table

Implementation option	Description	Deployment instructions
Enterprise-scale foundation	<p>Enterprise-ready platform foundation with all the necessary shared services to support the full IT portfolio, where connectivity can be added later as needed.</p> <p>Design principles Design areas</p>	Deploy to Azure Readme: foundation Readme: Network topology (Virtual WAN) Readme: Network topology (hub-spoke)
Azure landing zones modular	<p>Modular approach using Bicep for deploying the core platform capabilities.</p>	Readme: Bicep modules
Enterprise-scale for small enterprises	<p>This reference implementation is meant for organizations that don't have a large IT team and do not require fine grained administration delegation models.</p>	Deploy to Azure Readme
Enterprise-scale for Azure Government	<p>Reference implementation that can be deployed to Azure Government Cloud.</p>	Deploy to Azure Readme
CAF enterprise-scale landing zone (Azure China 21Vianet regions)	<p>Reference implementation that can be deployed to Azure clouds in China.</p>	Deploy to Azure Deploy
Azure landing zones Terraform module	<p>Deploys an enterprise-ready platform foundation using Terraform. Use this option when managing your platform using Terraform and need to accelerate delivery of the recommended resource hierarchy and governance model. Shared services, network connectivity, and application workloads can be integrated into your deployment or managed independently.</p>	Readme
Microsoft Cloud for Sovereignty	<p>A sovereign landing zone uses the same code base as the Azure landing zone Bicep approach but has more orchestration and deployment automation capabilities. It also has Azure Policy initiatives and assignments to help meet sovereignty requirements for public-sector customers, partners, and independent software vendors (ISVs).</p>	Readme

Customize approach

[+] Expand table

Implementation option	Description	Deployment instructions
Migration landing zone	Deploys the basic foundation for migrating low risk assets. Design areas	Deploy
Foundation blueprint	Adds the minimum tools need to begin developing a governance strategy. Design areas	Deploy
Partner landing zones	Partners who provide offerings aligned to the Ready methodology of the Cloud Adoption Framework can provide their own customized implementation option. Design principles	Find a partner ↗

Next steps

To proceed, choose one of the implementation options shown in the preceding tables. Each option includes a link to deployment instructions and the specific design principles that guide implementation.

Feedback

Was this page helpful?

 Yes

 No

Start with Cloud Adoption Framework enterprise-scale landing zones

Article • 01/04/2024

ⓘ Important

The Azure landing zones **Implementation options** section of the Cloud Adoption Framework is undergoing a freshness update.

As part of this update, we will be revising the table of contents and article content, which will include a combination of refactoring and consolidation of several articles. An update will be posted on this page once the work is completed.

Visit the new "[Deployment options](#)" section of the [Azure Architecture Center](#) for the latest Azure landing zone implementation content, including platform and application landing zones.

The enterprise-scale architecture represents the strategic design path and target technical state, for your Azure environment. It will continue to evolve alongside the Azure platform and is defined by the various design decisions that your organization must make, to map your Azure journey.

Not all enterprises adopt Azure in the same way. The Cloud Adoption Framework for Azure enterprise-scale landing zone architecture varies between customers. The technical considerations and design recommendations of the enterprise-scale architecture might lead to different trade-offs, based on your organization's scenario. Some variation is expected, but if you follow the core recommendations, the resulting target architecture will set your organization on a path to scale sustainably.

Prescriptive guidance

The enterprise-scale architecture provides prescriptive guidance, coupled with best practices for your Azure control plane. It follows design principles across the critical design areas for an organization's Azure environment.

Qualifiers: Should I start with enterprise-scale?

The enterprise-scale architecture is modular by design. It allows you to start with a foundational landing zone control plane that supports your application portfolios,

whether the applications are being migrated or are newly developed and deployed to Azure. The architecture can scale alongside your business requirements, regardless of scale point.

Start with a Cloud Adoption Framework enterprise-scale landing zone

The enterprise-scale approach to construct landing zones includes three sets of assets to support cloud teams:

- [Design guidelines](#): Guide to the critical decisions that drive the design of the Cloud Adoption Framework for Azure enterprise-scale landing zone.
- [Architecture](#): Conceptual reference architecture that demonstrates design areas and best practices.
- [Implementations](#): Azure Resource Manager template of the architecture to accelerate adoption.

Community

This guide is primarily developed by Microsoft architects and the broader Cloud Solutions Unit technical community. This community actively updates this guide, sharing lessons learned during enterprise-scale adoption efforts.

This guide has the same design principles as the standard Ready methodology. It expands on those principles to integrate subjects, such as governance and security, earlier in the planning process. Expanding the standard process is necessary because of a few natural assumptions that can be made when an adoption effort requires large-scale enterprise changes.

Next steps

[Implement a Cloud Adoption Framework enterprise-scale landing zone](#)

Implement Cloud Adoption Framework enterprise-scale landing zones in Azure

Article • 01/04/2024

ⓘ Important

The Azure landing zones **Implementation options** section of the Cloud Adoption Framework is undergoing a freshness update.

As part of this update, we will be revising the table of contents and article content, which will include a combination of refactoring and consolidation of several articles. An update will be posted on this page once the work is completed.

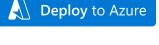
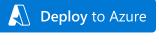
Visit the new "[Deployment options](#)" section of the [Azure Architecture Center](#) for the latest Azure landing zone implementation content, including platform and application landing zones.

Does your business require an initial implementation of landing zones? And do these landing zones need fully integrated governance, security, and an operations control plane from the start? With the following examples, you can use the Azure portal or infrastructure as code to set up and configure your Azure environment. It's also possible to transition between the portal and infrastructure as code (recommended) when your organization is ready.

Reference implementation

The following table lists example reference implementations based on the recommended enterprise-scale architecture.

ⓘ [Expand table](#)

Example deployment	Description	GitHub repo	Deploy to Azure
Enterprise-scale foundation	The suggested foundation for enterprise-scale adoption.	Example in GitHub	 Deploy to Azure
Enterprise-scale hub and spoke	Add a hub and spoke network module to the enterprise-scale foundation.	Example in GitHub	 Deploy to Azure

Example deployment	Description	GitHub repo	Deploy to Azure
Enterprise-scale Virtual WAN	Add a Virtual WAN network module to the enterprise-scale foundation.	Example in GitHub	 Deploy to Azure 
Enterprise-scale for small enterprises	Add a hub and spoke network architecture for small organizations.	Example in GitHub	 Deploy to Azure 
Enterprise-scale for Azure Government	Reference implementation that can be deployed to Azure Government and includes all options in a converged portal experience.	Example in GitHub	 Deploy to Azure 

Each reference implementation deploys platform resources to the selected target environment. Deployment details and an overview of the deployed resources can be found using the GitHub link in the table above.

Next steps

These examples provide an easy deployment option to support continued learning for the enterprise-scale approach. Before you use these examples in a production version of enterprise-scale, review the enterprise-scale architecture.

[Review the enterprise-scale architecture](#)

Scenario-specific enterprise-scale landing zones in Azure

Article • 01/04/2024

ⓘ Important

The Azure landing zones **Implementation options** section of the Cloud Adoption Framework is undergoing a freshness update.

As part of this update, we will be revising the table of contents and article content, which will include a combination of refactoring and consolidation of several articles. An update will be posted on this page once the work is completed.

Visit the new "[Deployment options](#)" section of the [Azure Architecture Center](#) for the latest Azure landing zone implementation content, including platform and application landing zones.

Scenarios for enterprise-scale landing zones enable effective adoption and operationalization of key technologies on Azure. These specific scenarios can be used to accelerate adoption when your organization identifies business or technical requirements that require additional tier-1 technology platforms to support mission-critical workloads or business processes.

Scenarios for enterprise-scale landing zones provide specific, refined deployments which can be run on top of existing enterprise-scale landing zones.

Use the following information to prepare your landing zones for mission-critical technology platforms and any supported workloads.

Scenarios

These landing zones have been developed in the broader context of common scenarios, such as:

[+] Expand table

Scenario	Description	Landing zone
Hybrid and multicloud	Guidance on shaping an organizations approach to implementing a hybrid cloud strategy.	Enterprise-scale for hybrid with Azure Arc

Scenario	Description	Landing zone
SAP	Guidance for migrating or adopting SAP workloads as part of your cloud strategy.	SAP on Azure landing zone accelerator
Virtual desktop	Guidance for migrating virtual desktops, or creating new as part of a cloud-focused productivity strategy.	Enterprise-scale for Azure Virtual Desktop
Modern application platform	Guidance for how application services and containers can be integrated into your cloud adoption strategy.	Enterprise-scale for AKS
Azure VMware Solution	Guidance for migrating VMware workloads to Azure as part of your cloud strategy.	Azure VMware Solution landing zone accelerator

Next steps

Scenario-specific enterprise-scale landing zones provide an accelerated path to building out your scalable, secure landing zones for various workloads. Select one of the links above to get started.

Azure landing zones Terraform module

Article • 05/22/2023

Azure provides native services for building your Azure landing zones. Other tools can also help with this effort. One tool that customers and partners often use to deploy landing zones is [Terraform by HashiCorp](#).

Deployment of resources to application landing zones is outside the scope of the module. Decisions on the deployment method and tooling are for the team that's responsible for the application.

The [Azure landing zones Terraform module](#) provides a rapid implementation of the platform resources that you need to manage [Azure landing zones](#) at scale by using Terraform. The module is designed to simplify the deployment of the management group hierarchy, policies, and resources in the connectivity and management subscriptions.

Prerequisites

If you're new to Terraform and you want information about installing and using it, see the [Install Terraform](#) tutorial on HashiCorp Learn.

For information on how to set up the Terraform provider and authenticate with Azure, see the [AzureRM provider guides](#) on the Terraform website. To learn how to set up the provider for deploying across multiple subscriptions, see the [Provider Configuration](#) wiki page.

Importance of using standard modules

Reuse of components is a fundamental principle of infrastructure as code. Modules are instrumental in defining standards and consistency across resource deployment within and across environments.

The Azure landing zones Terraform module is published to the official [Terraform Registry](#) and is verified by HashiCorp.

Deploying the module from the Terraform Registry provides:

- An accelerated delivery of Azure landing zones in your environment.
- A tested upgrade path to the latest version of the module, along with strict version control.

Benefits of using the module

Benefits of using the Azure landing zones Terraform module include:

- A managed and extensible core resource hierarchy for subscription organization through management groups.
- Scalable security governance and compliance through Azure identity and access management (IAM) controls, with an extensive library of custom definitions ready to assign.
- Enforcement of policy across subscriptions through management group inheritance.
- Managed resources for management and connectivity landing zones. These resources provide:
 - Assured policy compliance through tight integration of resources managed by the module and corresponding policy assignments.

- Integration between resources to reduce management overhead and provide an improved user experience, like automatic creation of virtual network links for Azure Private DNS.

💡 Tip

The template library is updated programmatically from the [Azure/Enterprise-Scale](#) GitHub repository. To stay up to date with the latest archetype configuration, policies, and roles, make sure you're using the latest version of the module.

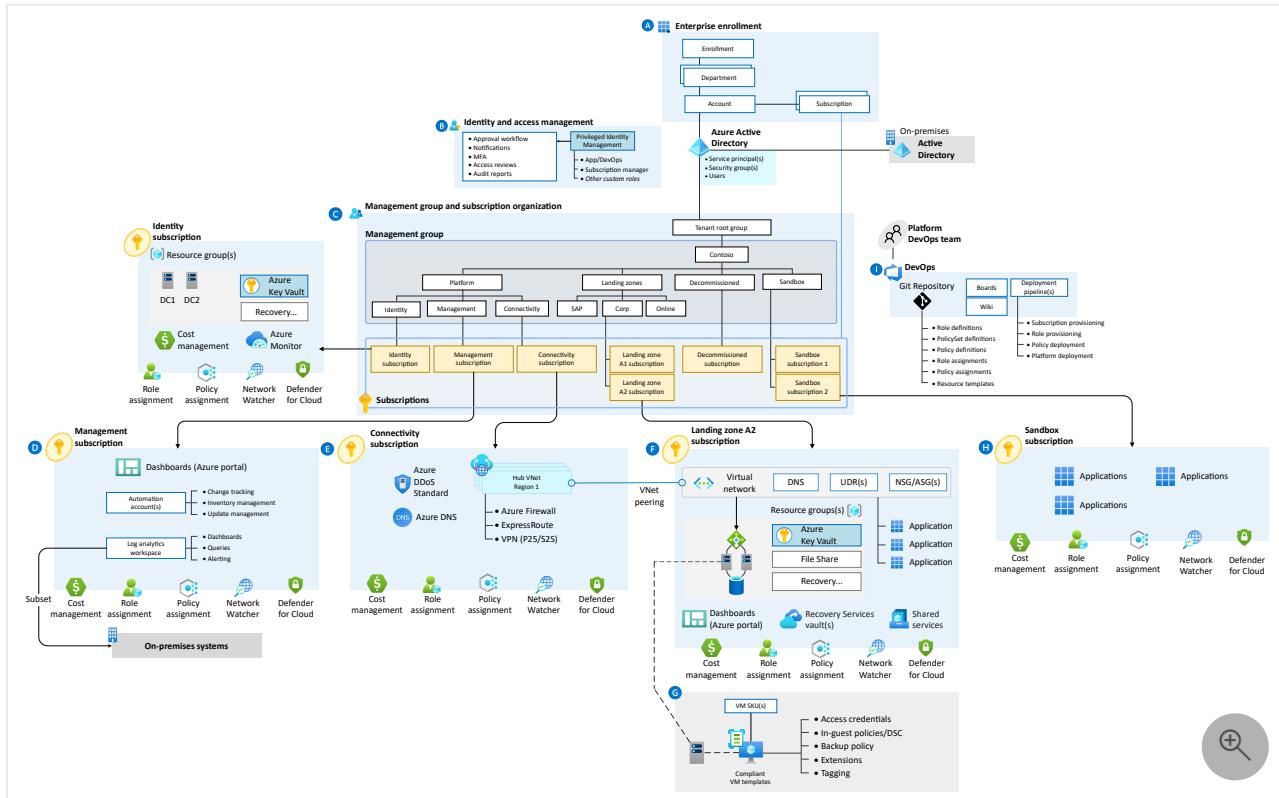
Resource deployment

You can configure the module to deploy sets of resources that align with the following critical design areas in Azure landing zones. Customize these resources to meet the requirements of your organization.

Resource category	Critical design area
Core resources	Resource organization Security Governance
Management resources	Management and monitoring
Connectivity resources	Network topology and connectivity
Identity resources	Identity and access management

Packaging these capabilities into a single Terraform module makes it easier to build and enforce consistency across the Azure platform when you're operating at scale.

These resources align with the [Azure landing zones conceptual architecture](#):

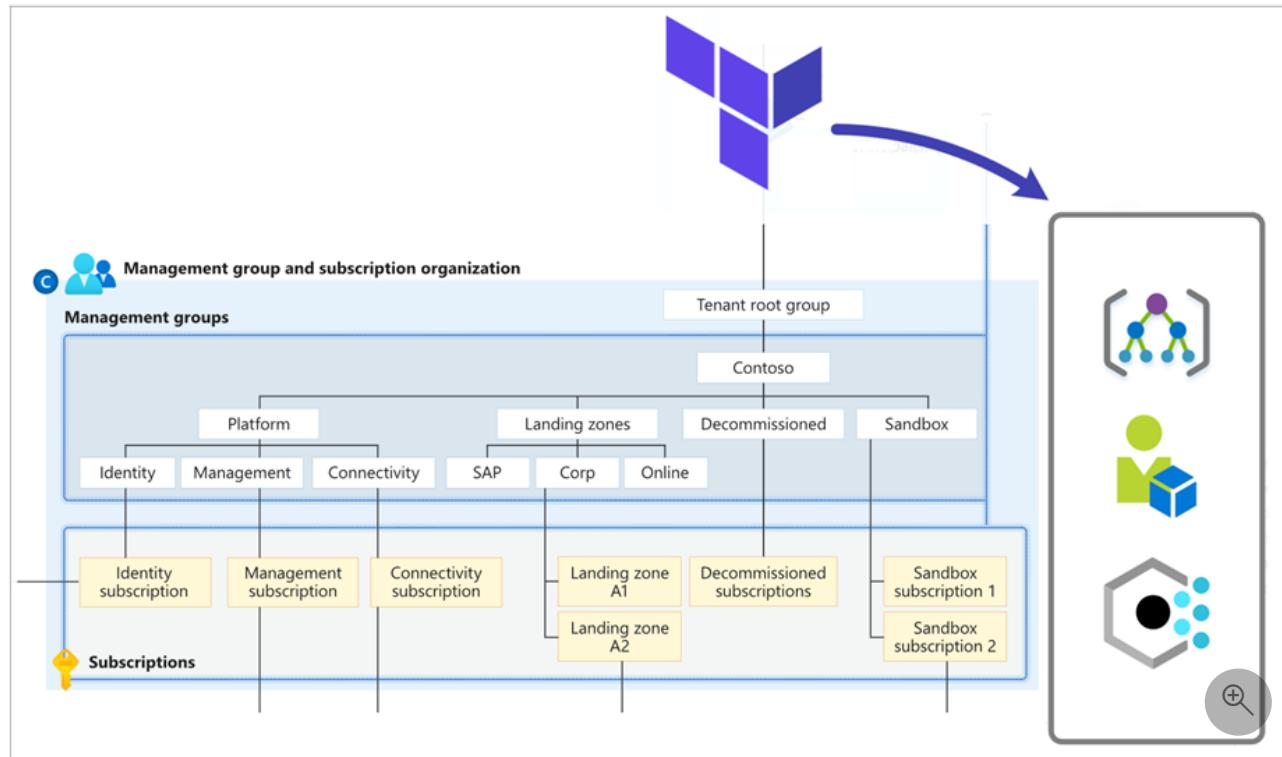


You can deploy these resources, by capability, across multiple subscriptions by using the [Provider Configuration](#) on the module block.

The following sections outline the resource types and configuration options.

Core resources

The core capability of this module deploys the foundations of the [conceptual architecture for Azure landing zones](#), with a focus on the central [resource organization](#).



When you enable deployment of core resources (*enabled by default*), the module deploys and manages the following resource types:

Resource	Azure resource type	Terraform resource type
Management groups	Microsoft.Management/managementGroups	azurerm_management_group
Management group subscriptions	Microsoft.Management/managementGroups/subscriptions	azurerm_management_group
Policy assignments	Microsoft.Authorization/policyAssignments	azurerm_management_group_policy_assignment
Policy definitions	Microsoft.Authorization/policyDefinitions	azurerm_policy_definition
Policy set definitions	Microsoft.Authorization/policySetDefinitions	azurerm_policy_set_definition
Role assignments	Microsoft.Authorization/roleAssignments	azurerm_role_assignment

Resource	Azure resource type	Terraform resource type
Role definitions	Microsoft.Authorization/roleDefinitions	azurerm_role_definition ↗

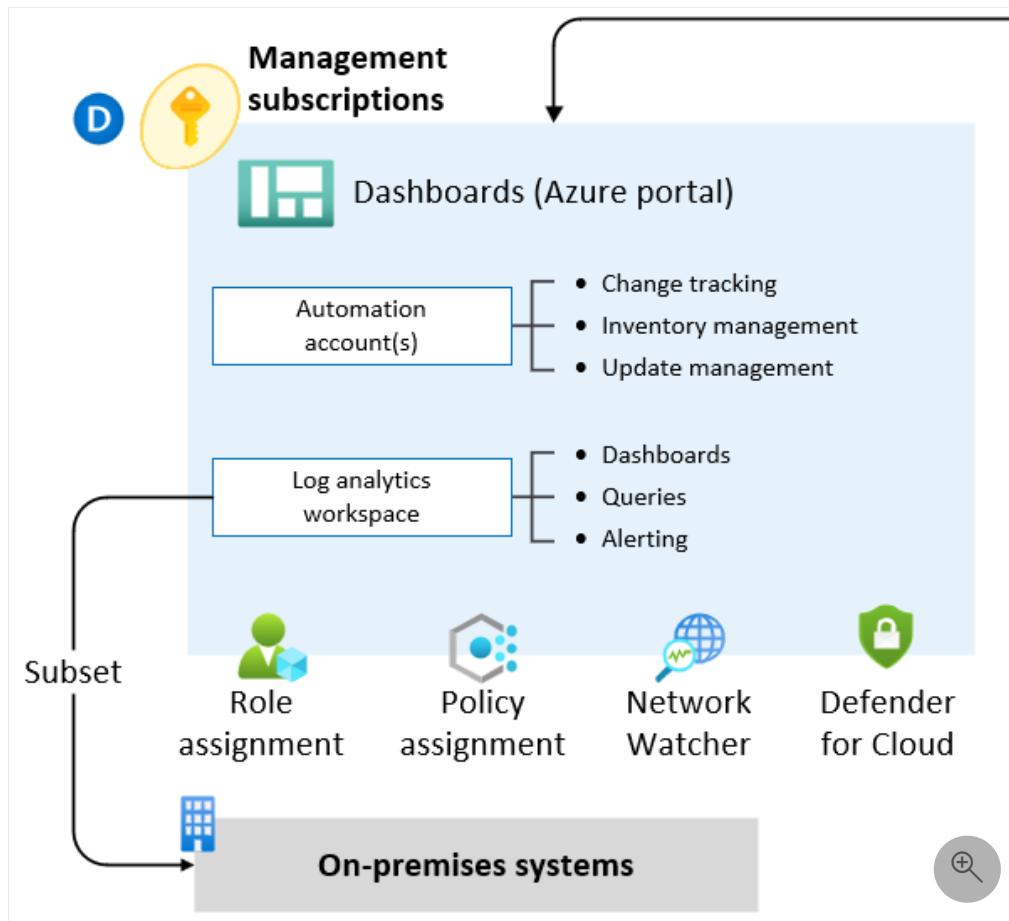
The exact number of resources that the module creates depends on the module configuration. For a [default configuration](#), you can expect the module to create approximately 180 resources.

💡 Tip

None of these resources are deployed at the subscription scope, but Terraform still requires a subscription to establish an authenticated session with Azure. For more information on authenticating with Azure, see the [Azure Provider: Authenticating to Azure \[↗\]\(#\)](#) documentation.

Management resources

The module provides an option to enable deployment of [management and monitoring](#) resources from the [conceptual architecture for Azure landing zones](#) into the specified subscription, as described on the [Provider Configuration \[↗\]\(#\)](#) wiki page. The module also ensures that the specified subscription is placed in the right management group.



When you enable deployment of management resources, the module deploys and manages the following resource types (*depending on configuration*):

Resource	Azure resource type	Terraform resource type
Resource groups	Microsoft.Resources/resourceGroups	azurerm_resource_group ↗

Resource	Azure resource type	Terraform resource type
Log Analytics workspace	Microsoft.OperationalInsights/workspaces	azurerm_log_analytics_workspace ↗
Log Analytics solutions	Microsoft.OperationsManagement/solutions	azurerm_log_analytics_solution ↗
Automation account	Microsoft.Automation/automationAccounts	azurerm_automation_account ↗
Log Analytics linked service	Microsoft.OperationalInsights/workspaces /linkedServices	azurerm_log_analytics_linked_service ↗

In addition to deploying the above resources, the module provides native integration into the corresponding policy assignments to ensure full policy compliance.

For more information about how to use this capability, see the [Deploy Management Resources \[↗\]\(#\)](#) wiki page.

Connectivity resources

The module provides an option to enable deployment of [network topology and connectivity](#) resources from the [conceptual architecture for Azure landing zones](#) into the current subscription context. It also ensures that the specified subscription is placed in the right management group.

This capability enables deployment of multiple hub networks based on any combination of [traditional Azure networking topology \(hub and spoke\)](#), and [Virtual WAN network topology \(Microsoft-managed\)](#).

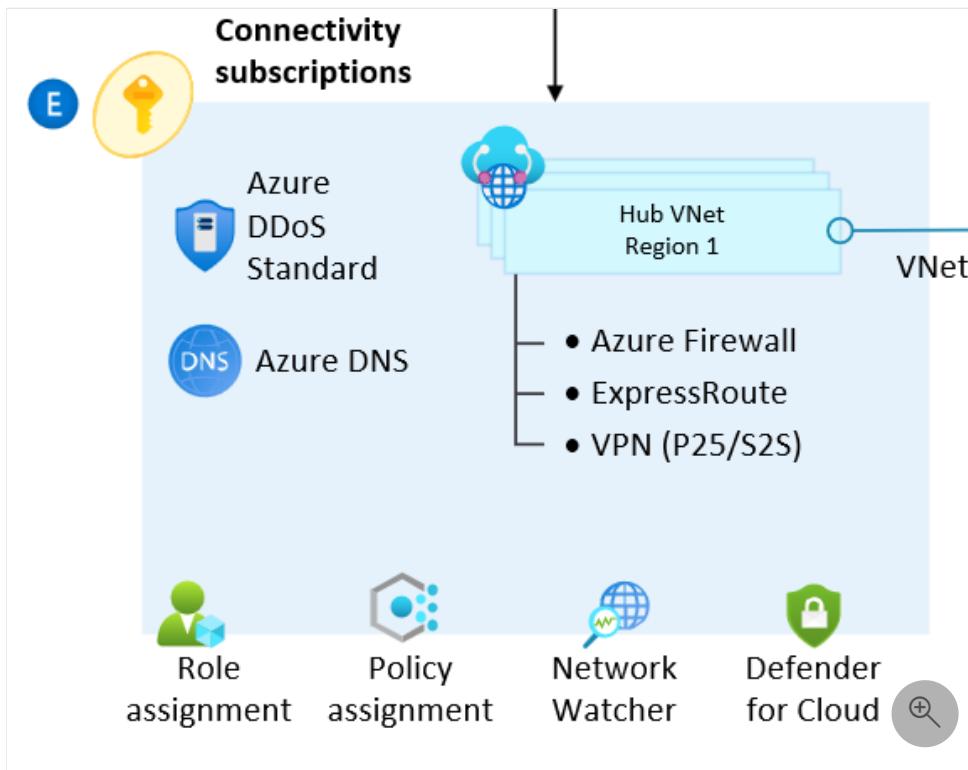
The module can also create and link [DDoS Network Protection](#) to Virtual Networks, and manage centralized public and private [DNS zones](#).

Note

We don't currently recommend DDoS IP Protection in Azure Landing Zones and recommend using this option in specific circumstances. Review the product documentation [About Azure DDoS Protection SKU Comparison](#)

Traditional Azure networking topology (hub and spoke)

The module can optionally deploy one or more hub networks based on the [traditional Azure networking topology \(hub and spoke\)](#).



① Note

The module currently configures only the networking hub and dependent resources for the connectivity subscription. Although there's an option to enable outbound virtual network peering from hub to spoke, users still need to initiate peering from spoke to hub. This is due to limitations in how the AzureRM provider targets a specific subscription for deployment.

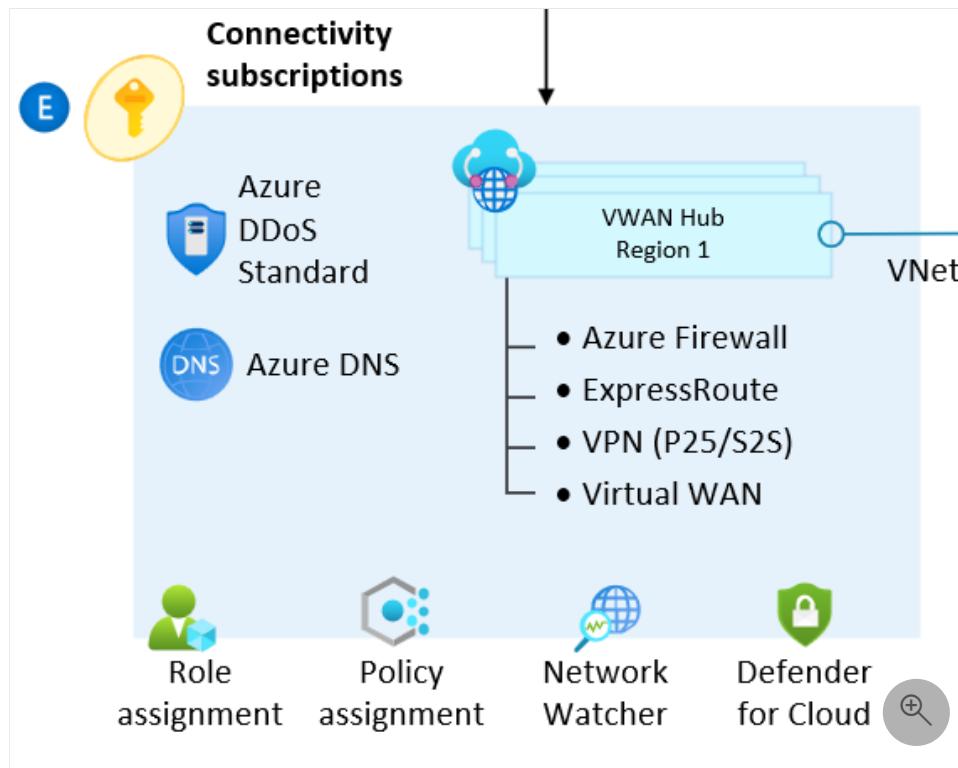
When you deploy resources based on a traditional Azure networking topology (hub and spoke), the module deploys and manages the following resource types (*depending on configuration*):

Resource	Azure resource type	Terraform resource type
Resource groups	Microsoft.Resources/resourceGroups	azurerm_resource_group ↗
Virtual networks	Microsoft.Network/virtualNetworks	azurerm_virtual_network ↗
Subnets	Microsoft.Network/virtualNetworks/subnets	azurerm_subnet ↗
Virtual network gateways	Microsoft.Network/virtualNetworkGateways	azurerm_virtual_network_gateway ↗
Azure firewalls	Microsoft.Network/azureFirewalls	azurerm_firewall ↗
Public IP addresses	Microsoft.Network/publicIPAddresses	azurerm_public_ip ↗
Virtual network peerings	Microsoft.Network/virtualNetworks/virtualNetworkPeerings	azurerm_virtual_network_peering ↗

For more information about how to use this capability, see the [Deploy Connectivity Resources](#) ↗ wiki page.

Virtual WAN network topology (Microsoft-managed)

The module can optionally deploy one or more hub networks based on the [Virtual WAN network topology \(Microsoft-managed\)](#).



① Note

Due to the different capabilities of Virtual WAN network resources over traditional peering for Virtual WAN spokes is bi-directional when using this capability.

When you deploy resources based on a Virtual WAN network topology (Microsoft-managed), the module deploys and manages the following resource types (*depending on configuration*):

Resource	Azure resource type	Terraform resource type
Resource Groups	Microsoft.Resources/resourceGroups	<code>azurerm_resource_group</code> ↗
Virtual WANS	Microsoft.Network/virtualWans	<code>azurerm_virtual_wan</code> ↗
Virtual Hubs	Microsoft.Network/virtualHubs	<code>azurerm_virtual_hub</code> ↗
Express Route Gateways	Microsoft.Network/expressRouteGateways	<code>azurerm_express_route_gateway</code> ↗
VPN Gateways	Microsoft.Network/vpnGateways	<code>azurerm_vpn_gateway</code> ↗
Azure Firewalls	Microsoft.Network/azureFirewalls	<code>azurerm_firewall</code> ↗
Azure Firewall Policies	Microsoft.Network/firewallPolicies	<code>azurerm_firewall_policy</code> ↗
Virtual Hub Connections	Microsoft.Network/virtualHubs/hubVirtualNetworkConnections	<code>azurerm_virtual_hub_connection</code> ↗

For more information about how to use this capability, see the [Deploy Virtual WAN Resources](#) ↗ wiki page.

DDoS Protection plan

The module can optionally deploy [DDoS Network Protection](#), and link Virtual Networks to the plan if needed.

ⓘ Note

Due to platform limitations, DDoS Protection plans can only be enabled for traditional virtual networks. Virtual Hub support is not currently available.

ⓘ Important

The Azure landing zones guidance recommends enabling DDoS Network Protection to increase protection of your Azure platform. To prevent unexpected costs in non-production and MVP deployments, this capability is disabled in the Azure landing zones Terraform module due to the cost associated with this resource.

For production environments, we strongly recommend enabling this capability.

When you enable deployment of deployment of DDoS Protection plan resources, the module deploys and manages the following resource types (*depending on configuration*):

Resource	Azure resource type	Terraform resource type
Resource groups	Microsoft.Resources/resourceGroups	azurerm_resource_group ↗
DDoS Protection plans	Microsoft.Network/ddosProtectionPlans	azurerm_network_ddos_protection_plan ↗

DNS

The module can optionally deploy [Private DNS zones to support Private Endpoints](#) and link them to hub and/or spoke Virtual Networks. User-specified public and private DNS zones can also be deployed and linked as needed.

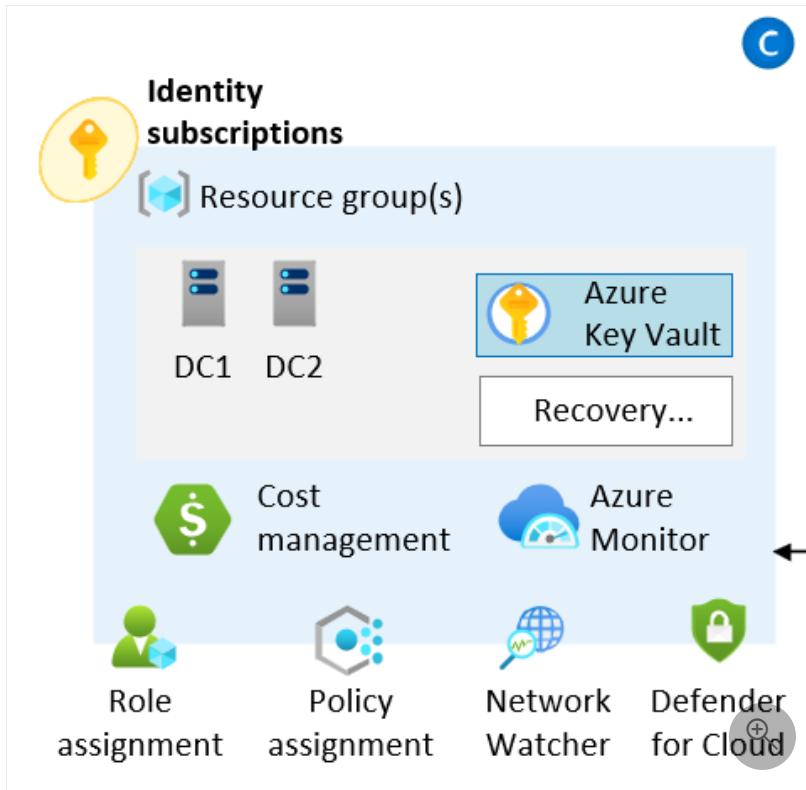
When you enable deployment of deployment of DNS resources, the module deploys and manages the following resource types (*depending on configuration*):

Resource	Azure resource type	Terraform resource type
Resource Groups	Microsoft.Resources/resourceGroups	azurerm_resource_group ↗
DNS Zones	Microsoft.Network/dnsZones	azurerm_dns_zone ↗
Private DNS Zones	Microsoft.Network/privateDnsZones	azurerm_private_dns_zone ↗

Resource	Azure resource type	Terraform resource type
Private DNS Zone Virtual Network Link	Microsoft.Network/privateDnszones/virtualnetworklinks	azurerm_private_dns_zone_virtual_network_link

Identity resources

The module provides an option to configure policies relating to the [identity and access management](#) landing zone. It also ensures that the specified subscription is placed in the right management group.



ⓘ Note

This capability doesn't deploy any resources. If you want to update policy settings related to the identity management group, use the `configure_identity_resources` input variable.

For more information about how to use this capability, see the [Deploy Identity Resources](#) wiki page.

Getting started

Requirements for getting started with the module are documented on the [Getting Started](#) Wiki page.

To simplify getting started, the module has been published to the [Terraform Registry](#). You can reference it directly within your code, as shown in the [simple example](#) later in this article. Running `terraform init` will automatically download the module and all dependencies.

You can view the latest module and provider dependencies on the [Dependencies](#) tab in the Terraform Registry.

ⓘ Important

There are known issues with some Terraform and AzureRM provider version combinations. You can resolve some known issues by upgrading to the latest Terraform and AzureRM provider versions. Other known issues are transient errors that you can typically fix by rerunning your deployment.

We generally recommend pinning to specific versions, and testing thoroughly before upgrading.

We'll release new versions of the module when changes are needed. New releases will ensure compatibility with the latest Terraform and AzureRM provider versions. Please refer to our [Module releases](#) guidance for more information.

To get the latest features, ensure that the module version is set to the latest version. If you're upgrading to a later version of the module, run `terraform init -upgrade`.

ⓘ release v4.0.2

Simple example

This example code deploys the minimum recommended [management group and subscription organization](#) from the enterprise-scale reference architecture. After you have this simple example up and running, you can start to customize your deployment.

ⓘ Tip

Even though `root_parent_id` is the module's only mandatory variable, we also recommend setting `root_id`. Changing the `root_id` value will start a full redeployment of all resources that the module manages, including downstream dependencies.

The following code is a simple starting configuration for your `main.tf` root module:

HashiCorp Configuration Language

```
# Configure Terraform to set the required AzureRM provider
# version and features{} block.

terraform {
  required_providers {
    azurerm = {
      source  = "hashicorp/azurerm"
      version = ">= 2.77.0"
    }
  }
}

provider "azurerm" {
  features {}
}

# Get the current client configuration from the AzureRM provider.
# This configuration is used to populate the root_parent_id variable with the
# current tenant ID used as the ID for the "Tenant Root Group"
# management group.
```

```

data "azurerm_client_config" "core" {}

# Use variables to customize the deployment

variable "root_id" {
  type    = string
  default = "es"
}

variable "root_name" {
  type    = string
  default = "Enterprise-Scale"
}

# Declare the Terraform Module for Cloud Adoption Framework
# Enterprise-scale and provide a base configuration.

module "enterprise_scale" {
  source  = "Azure/caf-enterprise-scale/azurerm"
  version = ">= 1.0.0"

  providers = {
    azurerm           = azurerm
    azurerm.connectivity = azurerm
    azurerm.management = azurerm
  }

  root_parent_id = data.azurerm_client_config.core.tenant_id
  root_id        = var.root_id
  root_name      = var.root_name
}

```

Next steps

The [Terraform module for Cloud Adoption Framework Enterprise-scale](#) provides an accelerated path to building out your enterprise-scale landing zones. It also provides the flexibility to expand and customize your deployment while maintaining a simplified approach to managing the configuration of each landing zone.

To find out more, [review the module on the Terraform Registry](#) and explore the [module documentation](#) on GitHub. In the documentation, you find more examples and tutorials about how to customize your deployment.

Learn how to [deploy the Azure landing zones Terraform module](#) through HashiCorp Learn. From there, you can also discover how some parts of the module work.

Independent software vendor (ISV) considerations for Azure landing zones

Article • 07/30/2024

For many organizations, the [Azure landing zones](#) conceptual architecture represents the destination of their cloud adoption journey. The landing zones describe how to build an Azure environment with multiple subscriptions. Each landing zone accounts for scale, security, governance, networking, and identity, and is based on feedback and lessons learned from many customers.

💡 Tip

It can be helpful to think of Azure landing zones as being like city plans. The architectures of workloads deployed into a landing zone are like plans for buildings in a city.

A city's water, gas, electricity, and transport systems all must be in place before buildings can be constructed. Similarly, an Azure landing zone's components, including management groups, policies, subscriptions, and role-based access control (RBAC), all must be in place before any production workloads can be deployed.

As an independent software vendor (ISV) building and operating your solution on Azure, you should refer to the following resources as you build your Azure environment:

- [Azure landing zones](#): Provides guidance for your overall Azure environment.
- [Azure Well-Architected Framework](#): Provides architectural guidance applicable to all workloads.
- [Architecting multitenant solutions on Azure](#): Provides specific architectural guidance for **multitenant** solutions on Azure.

The Azure landing zones help you choose a direction for your overall Azure environment. But as an ISV, SaaS provider, or startup, your specific implementation needs might differ from more standard customer scenarios. The following are just a few different implementation scenario examples:

- You build software that customers deploy into their own subscriptions.
- You have your own [control plane](#) and use automation scripts or software to deploy and configure Azure resources for your SaaS solutions.

- You're a small ISV or startup and want to start with the lowest possible cost, and might not want to initially use services like Azure Firewall and Azure DDoS Protection.
- You're a large SaaS ISV and plan to split your SaaS application across multiple subscriptions for scale. You also want to group the subscriptions so they correspond to your development, test, staging, and production environments.
- Your organization's operating model separates the roles of your corporate IT team and your SaaS product teams. Your organization's corporate IT team might manage resources like Microsoft Office 365 and Microsoft Teams, and your SaaS product team might be responsible for building and operating your SaaS product (including its central platform and identity components).

 **Note**

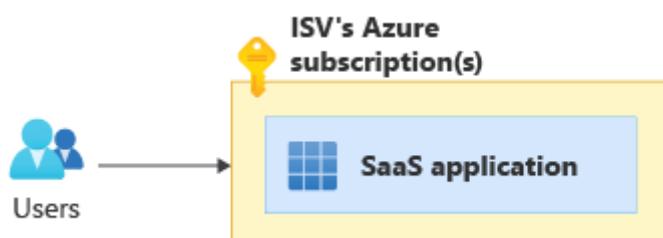
Sometimes, ISVs want to start with just a single Azure subscription that includes both platform "shared services" aspects and actual workload resources. Although this is technically possible, you'll face challenges later on when you need to move resources between subscriptions and find that not all [resource types can be moved](#). Review the [impact of design deviations](#) to understand what deviations are possible and their various levels of risk.

ISV deployment models

ISV solutions often fit into one of three deployment models: pure SaaS, customer-deployed, or dual-deployment SaaS. This section describes each model's different considerations for Azure landing zones.

Pure SaaS

In the pure SaaS model, your software is deployed fully only in your Azure subscriptions. End customers consume your software without deploying it in their own Azure subscriptions. In the following diagram, users are using a pure SaaS application provided by an ISV:



Examples of pure SaaS software include email-as-a-service, Kafka-as-a-service, cloud-data-warehouse-as-a-service, and many [SaaS listings in Azure Marketplace](#).

If you're a small SaaS ISV, you might not need to use multiple Azure subscriptions to deploy your resources right away. But as you scale, Azure's subscription limits can affect your ability to scale within a single subscription. Review the [enterprise-scale landing zone design principles](#), particularly subscription democratization, and familiarize yourself with the [architectural approaches for multitenancy](#) to plan for your future growth.

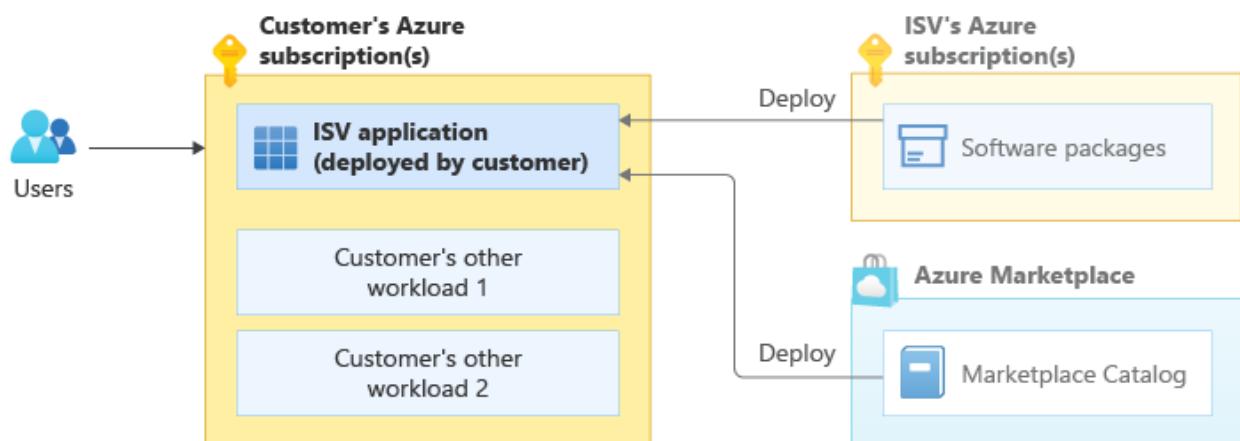
ISVs building pure SaaS solutions should consider the following questions:

- Should all the Azure resources that make up our SaaS solution be in one Azure subscription, or partitioned across multiple Azure subscriptions?
- Should we host each customer in their own dedicated Azure subscription, or can we create resources within one or a few shared subscriptions?
- How can we apply the [Deployment Stamp \(scale unit\) pattern](#) to all of our solution's tiers?
- How can we use [Azure resource organization in multitenant solutions](#) to keep us from facing scale challenges and Azure subscription limits?

Customer-deployed

In the customer-deployed model, your end customers purchase software from you and then deploy it into their own Azure subscriptions. They might initiate the deployment from the Azure Marketplace, or do it manually by following instructions and using scripts you provide.

In the following diagram, an ISV provides a software package or Azure Marketplace catalog product, and users deploy that resource into their own Azure subscriptions alongside their other workloads:



The *Customer's other workload* element in the diagram can represent either a customer's own workload or another ISV product the customer has deployed within their Azure

subscription. Customers frequently deploy multiple products from different ISVs into their Azure subscriptions. They combine these individual products to create solutions. For example, a customer might deploy a database product from one ISV, a network virtual appliance from another ISV, and a web application from a third ISV.

Examples of customer-deployed ISV products include the many [virtual machine images](#) (such as network and storage virtual appliances) and [Azure applications](#) in the Azure Marketplace.

For some customer-deployed solutions, an organization might provide management of and updates for the solution deployed within their end-customer Azure subscriptions by using [Azure Lighthouse](#) or [Azure Managed Applications](#). ISVs, Solution Integrators (SIs), and Managed Service Providers (MSPs) all can use this strategy when it meets their particular needs.

Customer-deployed ISV solutions are considered a standard application workload from the perspective of Azure landing zones. Consider the [Azure landing zones guidance](#) as you design your product to work with the [Azure landing zones design principles](#) your Azure customers adopt.

It's especially important for you to have a good understanding of the Azure landing zone concepts as you migrate your existing customers' workloads to Azure.

ISVs building customer-deployed solutions should consider the following questions:

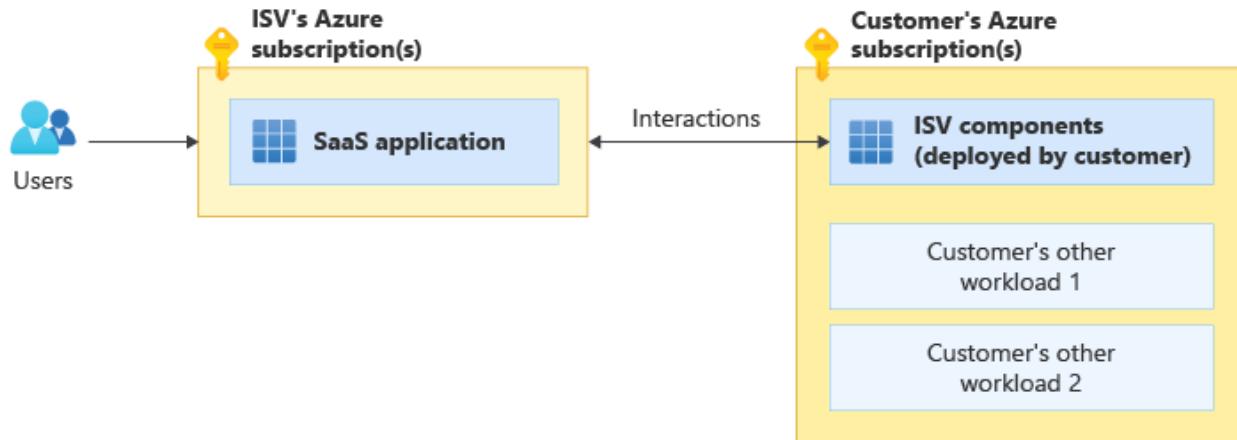
- Should a customer deploy our solution into its own dedicated subscription or into an existing subscription that contains related workloads?
- How should customers establish network connectivity between existing workloads (inside and outside of Azure) and our solution?
- Does our solution support authentication mechanisms from Microsoft Entra ID or require other protocols like LDAP or Kerberos?
- How do we reduce or eliminate Azure Policy violations, like those caused by conflicts between our solution templates and a customer's Azure policies?

Customer Azure policies that can cause Azure Policy violations include examples like "All subnets must have a network security group" and "No public IP addresses can be attached to network interfaces in the Corp landing zone". Keep the potential for these conflict-causing policies in mind as you plan your deployment.

Dual deployment SaaS

Some SaaS solutions interact with or use resources that are deployed in customers' Azure subscriptions. This deployment model is sometimes called *dual deployment SaaS*.

or *SaaS hybrid*. In the following diagram, an ISV provides a hosted SaaS solution that interacts with resources deployed into an end customer's Azure subscription:



A real-world example of *dual deployment SaaS* is Microsoft Power BI, a SaaS service that can optionally use a Power BI on-premises data gateway deployed on a virtual machine in a customer's Azure subscription.

Other examples of *dual deployment SaaS* scenarios include:

- Your organization builds Virtual Desktop Manager, a product that provides a SaaS console interface to control Azure Virtual Desktop resources in each customer's Azure subscription.
- Your organization provides a SaaS console for data analytics, and dynamically creates and deletes compute node virtual machines in each customer's Azure subscription.

As a dual deployment ISV, you should refer to the Azure landing zone for guidance in two areas: structuring your own Azure environment to host your SaaS service, and ensuring proper interaction between your deployments in customers' Azure subscriptions and your customers' landing zones.

ISVs building dual deployment SaaS solutions should consider the following questions:

- Have we reviewed all considerations for building both pure SaaS and customer-deployed solutions?
- Which components of our solution should be hosted in our Azure subscriptions, and which components should be customer-deployed?
- How can we ensure secure provisioning and interactions with resources deployed in our customers' Azure subscriptions?

Azure landing zone design principles and implementations

Azure's landing zone design principles recommend aligning with Azure-native platform capabilities such as Log Analytics, Azure Monitor, and Azure Firewall. The landing zone guidance also provides specific [Azure landing zone implementation options](#).

As an ISV, you might decide to implement your own landing zone environments. You might need to use your own automation to deploy Azure resources across subscriptions. Or you might want to continue using tools you already employ for logging, monitoring, and other platform-layer services.

If you do implement your own landing zone environments, we recommend that you use Azure landing zone guidance and sample implementations for reference, and align your approach with proven Azure landing zone designs.

Microsoft Entra tenants

Each Azure landing zone and its management group hierarchy is rooted in a single Microsoft Entra tenant. This means that the first decision you need to make is which Microsoft Entra tenant to use as the source of identities for managing your Azure resources. Identities in the Microsoft Entra ID include users, groups, and service principals.

Tip

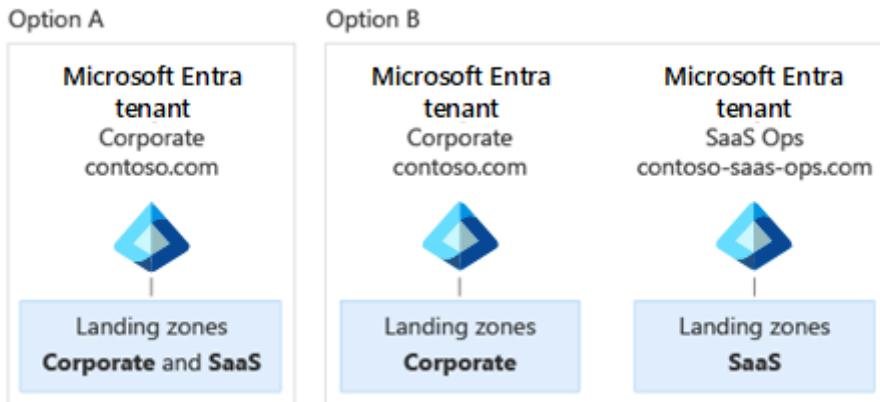
The Microsoft Entra tenant you select for your landing zone doesn't affect your application-level authentication. You can still use other identity providers like Azure AD B2C regardless of which tenant you choose.

The [guidance for Azure landing zones and Microsoft Entra tenants](#) strongly recommends using a single Microsoft Entra tenant, and this is the correct approach for most situations. However, as a SaaS ISV, you might have reason to use two tenants.

For some SaaS ISVs, one team manages corporate resources and a separate team operates the SaaS solution. This separation can be for operational reasons or to comply with regulatory requirements. Perhaps your corporate IT team isn't allowed to manage any SaaS-related subscriptions and resources, so they can't be administrators of the Microsoft Entra tenant. If this scenario applies to you, consider using two separate Microsoft Entra tenants: one tenant for corporate IT resources like Office 365, and one tenant for Azure resources that comprise your SaaS solution.

Each Microsoft Entra tenant must have its own domain name. If your organization uses two tenants, you might choose a name like `contoso.com` for your corporate Microsoft

Entra tenant and `contoso-saas-ops.com` for your SaaS Microsoft Entra tenant, as shown in the following diagram.



⚠️ Warning

When you use multiple Microsoft Entra tenants, your management overhead increases. If you use Microsoft Entra ID P1 or P2 features like Privileged Identity Management, you have to purchase individual licenses for each Microsoft Entra tenant. It's best to only use multiple Microsoft Entra tenants if your situation truly requires it.

Avoid using separate Microsoft Entra tenants for pre-production and production environments. Rather than creating two tenants like `contoso-saas-ops-preprod.com` and `contoso-saas-ops-prod.com` with separate Azure subscriptions under each, you should create one Microsoft Entra tenant. You can use management groups and Azure RBAC to govern the access to subscriptions and resources under this single tenant.

For more information on the using multiple Microsoft Entra tenants, see [Azure landing zones and multiple Microsoft Entra tenants](#) and [resource isolation with multiple tenants](#).

Management groups

The [Azure landing zone conceptual architecture](#) recommends using a specific management group hierarchy. However, ISVs can have different requirements than other organizations. This section describes some ways your ISV organization might choose to adopt different practices than what the landing zone conceptual architecture recommends.

Top-level management group

Your management group hierarchy is nested under the Azure-created **Tenant root group** management group. You don't use the **Tenant root group** directly.

A standard organization that has a centralized corporate IT team managing their platform and shared services (like logging, networking, identity, and security) usually creates one top-level management group under the Azure-created **Tenant root group** and deploys the rest of their management groups below it. This top-level management group is usually named after the organization itself (such as *Contoso*).

As a SaaS ISV, you might have one SaaS product or you might have a few separate SaaS products or lines of business. While you should generally use the same Microsoft Entra tenant to manage Azure resources across all of your products (as discussed in the [Microsoft Entra tenants](#) section), in some scenarios you might choose to deploy multiple management group hierarchies.

Consider how independent your products are from each other, and ask yourself:

- Do our products all use the same platforms for DevOps, identity, security, connectivity, and logging?
- Are those shared services operated by a single central team?

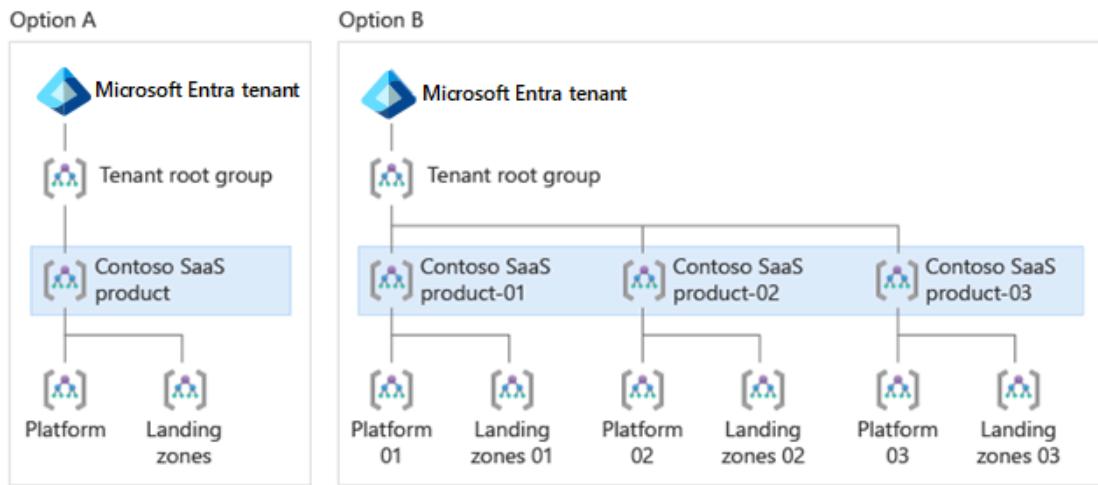
If you answered yes to both questions, create a single top-level **SaaS Product** management group under the **Tenant root group**.

If you instead answered *no*, and each of your SaaS products is managed and operated by separate platform teams, consider creating a separate top-level management group for each product, like the two top-level management groups **SaaS Product-01** and **SaaS Product-02**.

💡 Tip

It's uncommon for one ISV to have more than just a few top-level management groups. Often, several products can be combined together due to similarities in how they're managed and operated.

This management approach is similar to the [testing approach for enterprise-scale landing zones](#). However, rather than creating *Contoso* and *Contoso-Canary* under the **Tenant root group**, in this approach the example company would create the product-specific *Contoso-SaaS-Product-01*, *Contoso-SaaS-Product-02*, and *Contoso-SaaS-Product-03* top-level management groups under it instead. This scenario is illustrated in the following diagram:



Platform management group

In the [Azure landing zone resource organization hierarchy](#), the **Platform** management group contains all Azure subscriptions that host components and shared services used by workloads in the landing zone subscriptions. Examples of components deployed into the platform and shared services subscriptions include centralized logging infrastructure (such as Log Analytics workspaces), DevOps, security, automation tooling, central networking resources (such as hub-VNet and DDoS Protection plans), and an ISV's control plane services.

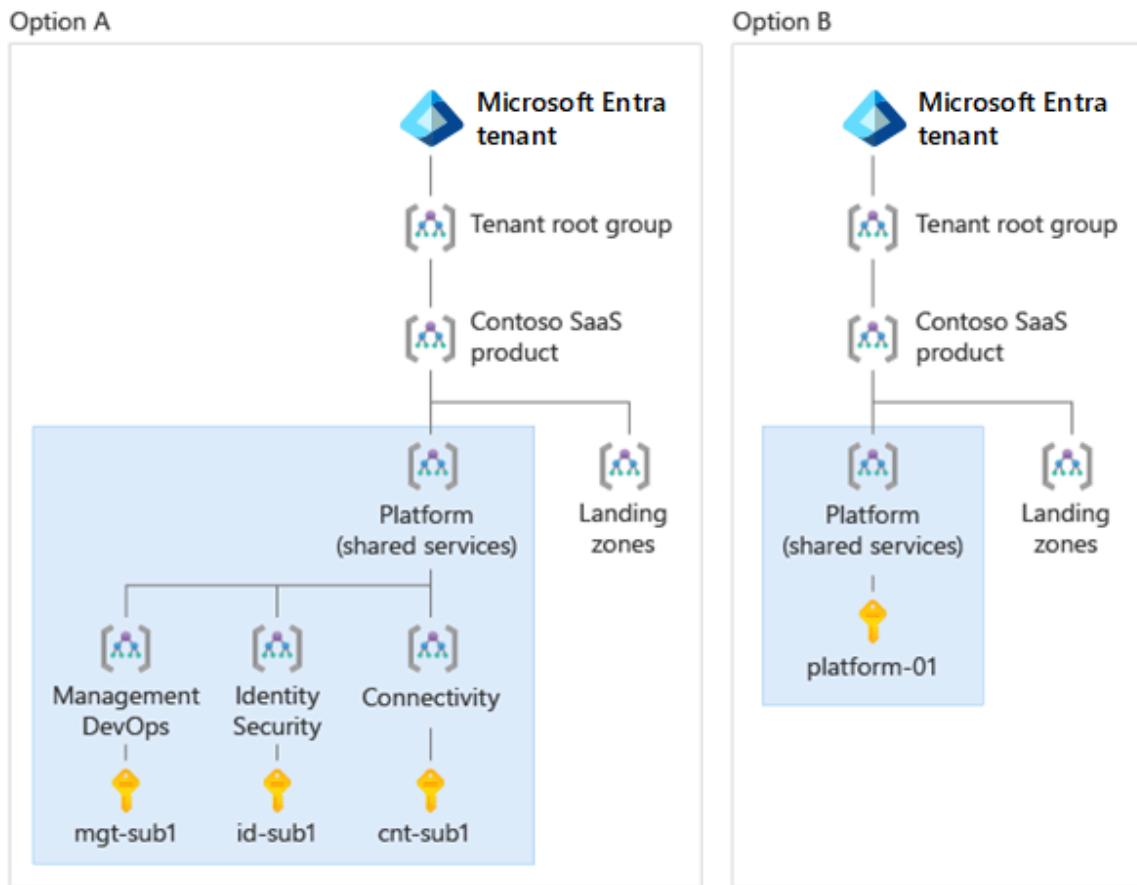
The **Platform** management group is frequently partitioned into **Identity**, **Management**, and **Connectivity** child groups to provide convenient separation of roles and policies for enterprise customers.

In your organization, you might have a single team that manages all shared platform components like identity, networking, and management. If so, and if you have no plans to separate that management across multiple teams, then consider using a single **Platform** management group.

If you instead will have separate teams that manage different parts of your centralized platform, you should deploy further levels in the management group hierarchy under the **Platform** management group. This allows you to assign separate policies for each part of your centralized platform.

The following diagram illustrates two potential implementations of the **Platform** management group. Option A shows a more comprehensive scenario, where the **Platform** management group contains three child management groups: **Management** and **DevOps**, **Identity and Security**, and **Connectivity**. Each child management group contains a subscription with the relevant resources. Option B shows a more simple

scenario, where the **Platform** management group contains a single platform subscription.



Landing Zones management group

The **Landing Zones** management group contains the Azure subscriptions that host your SaaS solution's actual subsystems and workloads.

This management group contains one or more child management groups. Each of the child management groups under **Landing Zones** represents a workload or subsystem *archetype*, with consistent policy and access requirements that should apply to all subscriptions. Reasons for using multiple archetypes include:

- **Compliance:** If a subsystem of your SaaS product needs to be PCI-DSS compliant, consider creating a **PCI DSS** archetype child management group under **Landing Zones**. All Azure subscriptions that contain resources within the scope of PCI-DSS compliance should be placed within that management group.
- **Tiers:** Consider creating separate landing zone archetypes for your SaaS solution's *dedicated* tier customers and *free* tier customers. Each of the child management groups contains different Azure Policy settings. For example, the policies in the free tier might restrict deployments to only enable specific virtual machine SKUs, and

the policies in the dedicated tier might require resources to be deployed into specific regions.

Environment-specific management groups

SaaS ISVs often organize their cloud environments by modeling their software development lifecycle environments in a sequence. This commonly requires deployment first to a *Development* environment, then to a *Test* environment, then to a *Staging* environment, and finally to a *Production* environment.

One common difference between the environments is their Azure RBAC rules, like who can access each group of subscriptions. For example, the DevOps, SaaSOps, development, and test teams might all have different levels of access to different environments.

Important

Most Azure customers have hundreds of applications and use separate Azure subscriptions for each application team. If each application had its own development, test, staging, and production management groups, there would be a large number of management groups with near-identical policies. For most customers, the [Enterprise-Scale Landing Zone FAQ](#) advises against using separate management groups for each environment. It recommends using separate subscriptions within a single management group instead.

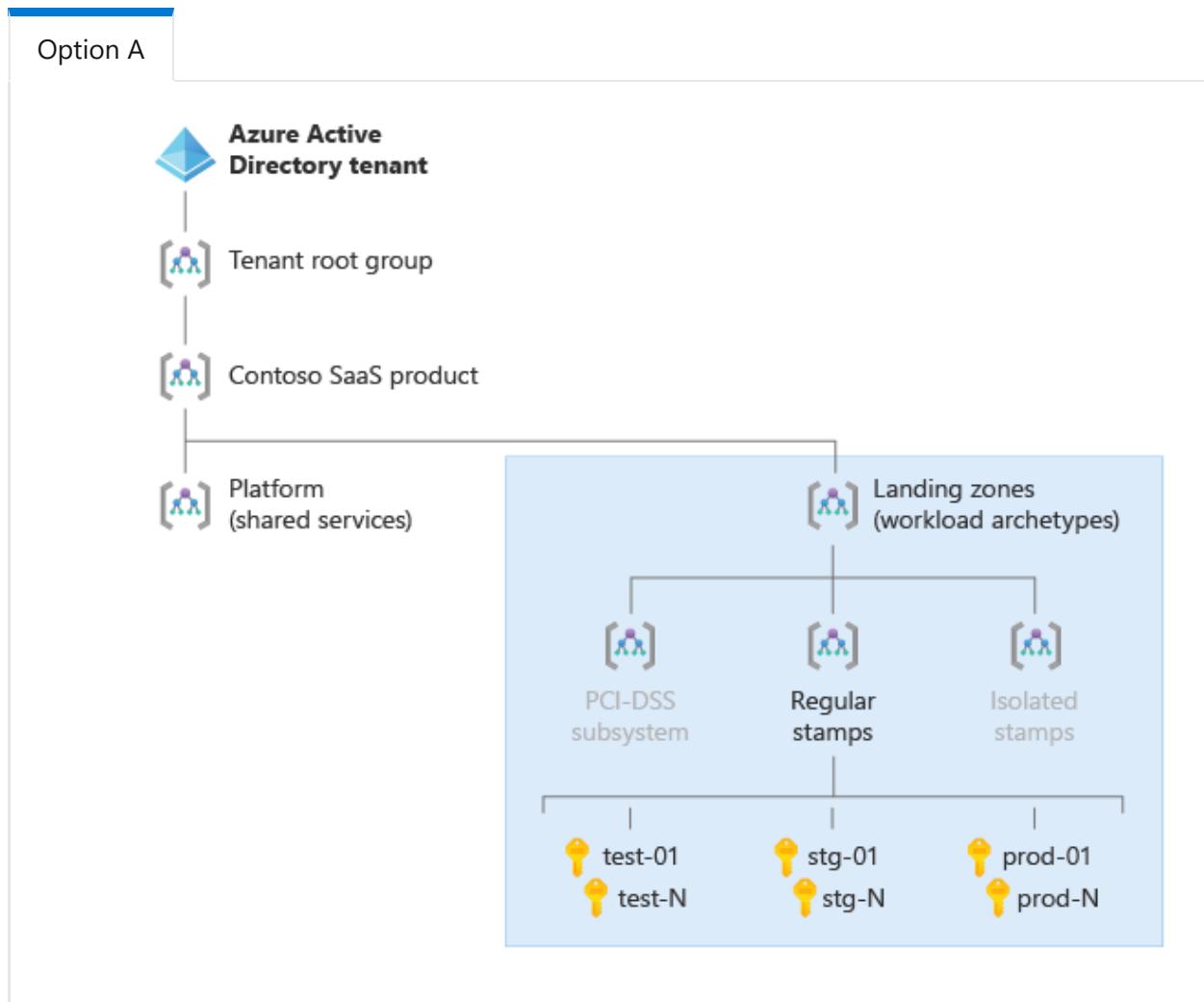
However, SaaS ISVs can have different requirements than most other Azure customers, and might have good reason to use environment-specific management groups in some situations.

SaaS ISVs sometimes need to group multiple subscriptions that represent *shards* or *partitions* of the same subsystem, application, or workload. You might need to apply policies or role assignments to groups of subscriptions in a noticeably different way than in the archetype management group. In this case, consider creating child management groups that correspond to each environment under the archetype management group.

The following diagrams illustrate two potential options. Option A shows a scenario with separate subscriptions for each environment but no environment-specific management groups. Option B shows a SaaS ISV scenario with environment-specific management groups under the **Regular stamps** management group. Each environment-specific management group contains multiple subscriptions. Over time, the ISV scales their

Azure resources in each environment across an increasing number of subscriptions with a common set of policies and role assignments.

Select each tab to see the two diagrams.



Decommissioned and Sandboxes management groups

The Azure landing zone [resource organization guidance](#) recommends including **Decommissioned** and **Sandboxes** management groups directly below your top-level management group.

The **Decommissioned** management group is a holding place for Azure subscriptions that are being disabled and will eventually be deleted. You can move a subscription that's no longer in use into this management group to track it until all the resources in the subscription are permanently deleted.

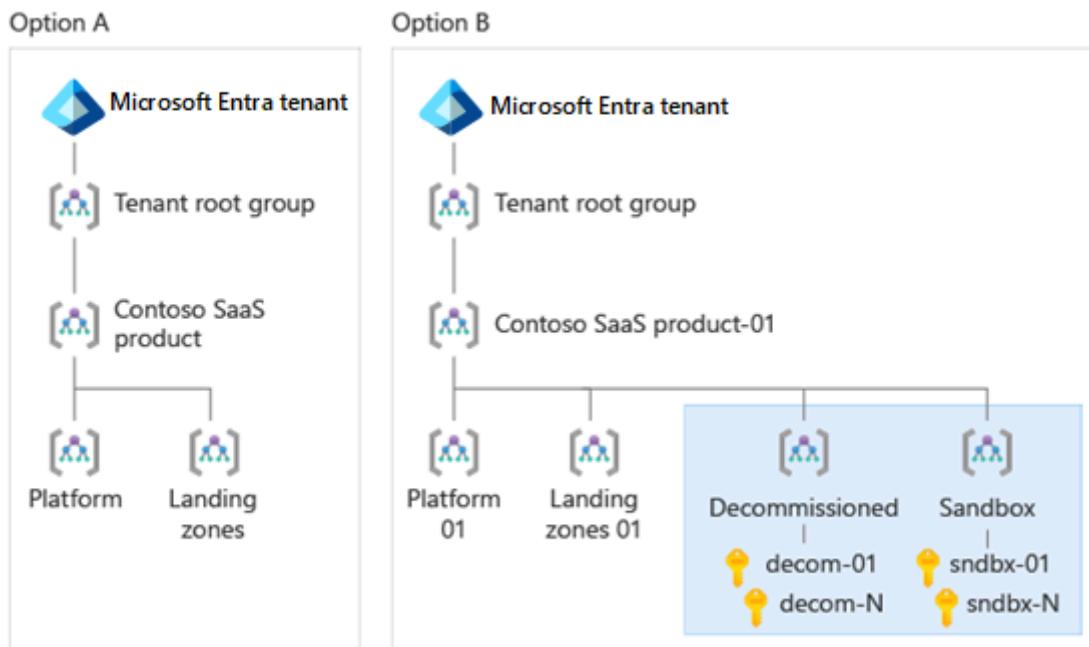
The **Sandboxes** management group usually contains [Azure subscriptions that are used for exploration purposes](#) and have loose or no policies applied to them. For example, you might provide individual developers with their own subscriptions for development and testing. You can avoid applying the normal policies and governance to these

subscriptions by placing them in the **Sandboxes** management group. This increases the developers' agility and enables them to easily experiment with Azure.

ⓘ Important

Subscriptions in the **Sandboxes** management group should not have [direct connectivity to the landing zone subscriptions](#). Avoid connecting sandbox subscriptions to production workloads or to any non-production environments that mirror production environments.

The following diagram illustrates two potential options. Option A doesn't include the **Decommissioned** and **Sandbox** management groups, while option B does.



Example ISV landing zones

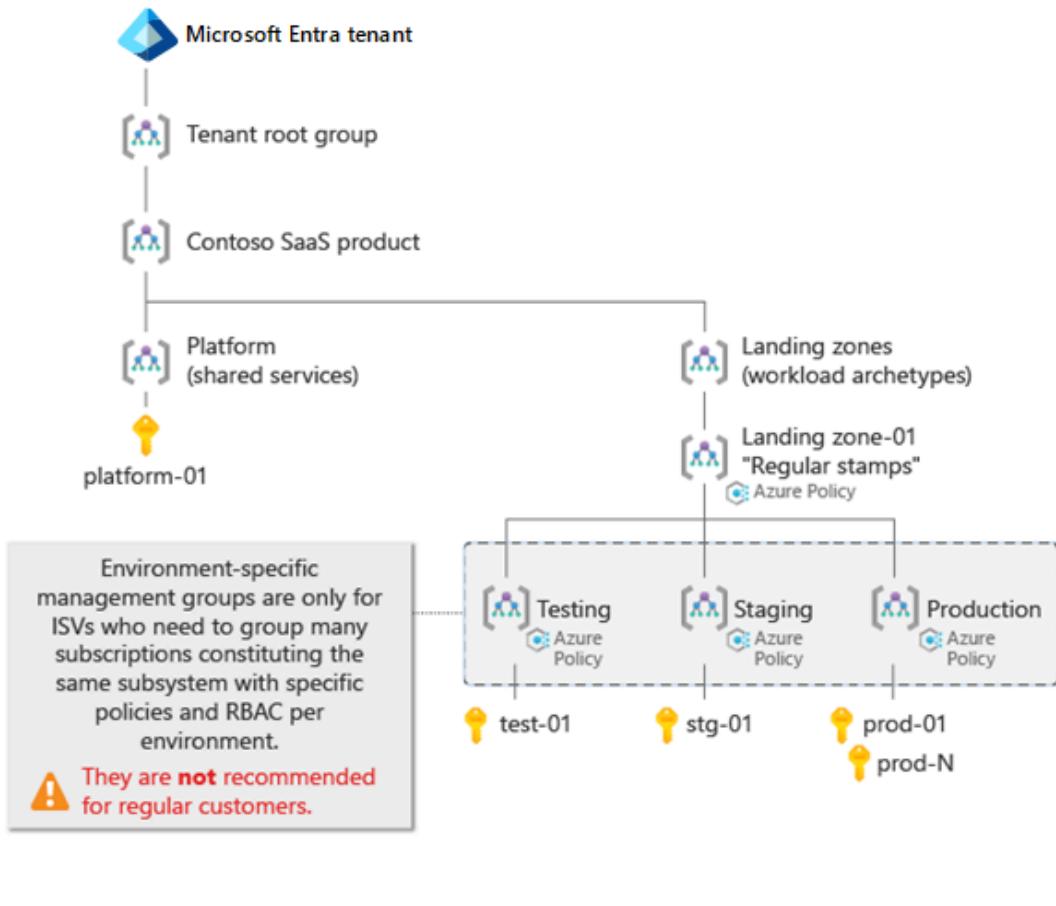
This section includes two example Azure landing zone structures for a SaaS ISV. Select each tab to compare the two example landing zones.

Minimal

The following diagram shows an example SaaS ISV Azure landing zones hierarchy with the following characteristics:

- The ISV keeps all their platform components in a single Azure subscription, instead of [splitting them into multiple platform management groups](#).
- There is only one [landing zone management group](#).

- The landing zone includes [environment-specific management groups](#) for organizing subscriptions and assigning different policies and roles.
- The ISV didn't include the management groups for [decommissioned and sandbox subscriptions](#).



Next steps

- If you're building a multitenant solution, learn more about [architecting multitenant solutions on Azure](#).
- Learn [what is an Azure landing zone](#).
- Learn about [Azure landing zone design areas](#).

Feedback

Was this page helpful?

Yes

No

Sovereignty considerations for Azure landing zones

Article • 11/07/2023

Adopting cloud computing while meeting digital sovereignty requirements is complex and can differ greatly between organizations, industries, and geographies. [Microsoft Cloud for Sovereignty](#)  addresses the sovereignty needs of government organizations by combining the power of the global Azure platform with several sovereignty capabilities that are designed to help mitigate sovereignty risks.

Microsoft Cloud for Sovereignty

Microsoft Cloud for Sovereignty provides capabilities across various layers:

- Advanced sovereign control services like Azure confidential computing and Azure Key Vault Managed Hardware Security Module (Managed HSM)
- Sovereign guardrails through codified architecture, workload accelerators, localized Azure Policy initiatives, tooling, and guidance
- Regulatory compliance and transparency into the cloud operator's activities
- Functionality that's built on top of the Azure public cloud capabilities



Public sector customers with sovereignty needs who want to start using Azure can benefit from Microsoft Cloud for Sovereignty. The tools and guidelines that Microsoft Cloud for Sovereignty provides, such as the sovereign landing zone (preview), can accelerate the definition and deployment of a sovereign environment.

Sovereign landing zone

The sovereign landing zone (preview) is an opinionated tailored variant of the [Azure landing zone architecture](#) that's intended for organizations that need advanced sovereignty controls. The sovereign landing zone (preview) aligns Azure capabilities such as service residency, customer-managed keys, Azure Private Link, and confidential computing to create a cloud architecture where data and workloads offer encryption and protection from threats by default.

ⓘ Note

Microsoft Cloud for Sovereignty is oriented toward government organizations with sovereignty needs. You should carefully consider whether you need the Microsoft Cloud for Sovereignty capabilities, and only then consider adopting the sovereign landing zone (preview) architecture.

Sovereign landing zone design areas

The Azure landing zone architecture consists of eight [design areas](#). Each design area describes factors to consider before you deploy a landing zone. The following sections describe additional considerations that apply when you deploy the sovereign landing zone (preview). Besides the Azure landing zone guidance, also keep these new considerations in mind.

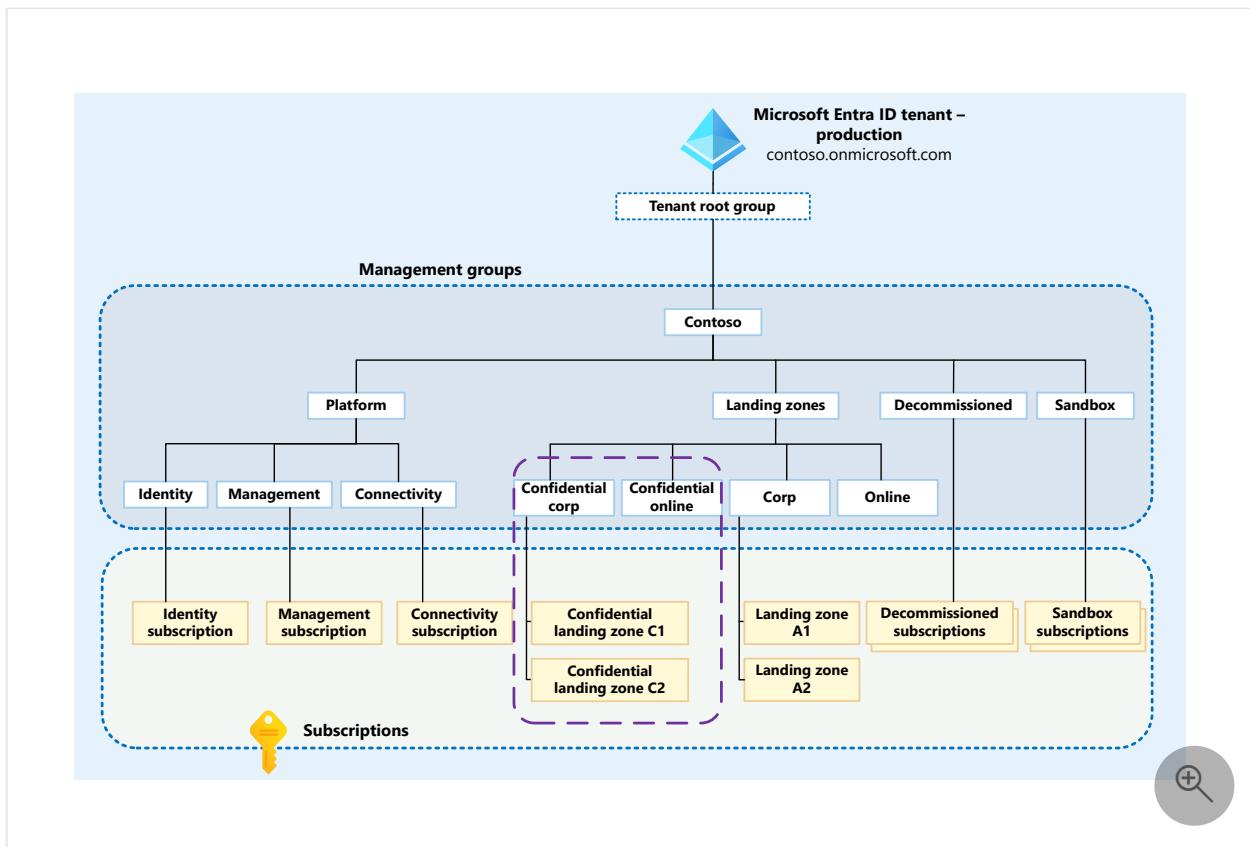
Resource organization

The sovereign landing zone is a tailored version of the Azure landing zone conceptual architecture. The sovereign landing zone aligns to the guidance that's outlined in [Tailor the Azure landing zone architecture](#).

Management groups for confidential computing

As the following diagram shows, the sovereign landing zone architecture builds on the Azure landing zone architecture:

- Under the *Landing zones* management group, the *Confidential corp* and *Confidential online* management groups are added.
- A set of specific policy initiatives, for example, [Microsoft Cloud for Sovereignty policy baseline](#), are also applied. These initiatives offer controls such as resource deployment location, resource deployment types, and encryption.



Microsoft Cloud for Sovereignty policy baseline

The sovereign landing zone (preview) comes with the Microsoft Cloud for Sovereignty policy baseline initiatives deployed. As a result, you can deploy other policy sets within the sovereign landing zone (preview). You can layer extra policies on top of the sovereign landing zone (preview). Examples include Azure landing zone policies and policy sets that address control frameworks such as National Institute of Standards and Technology (NIST) 800 171 Revision 2 and Microsoft Cloud Security Benchmark.

The Microsoft Cloud for Sovereignty policy baseline consists of:

- Policies to enforce the use of confidential computing resources when workloads are deployed into the confidential management groups. These policies help create a platform where workloads are protected at rest, in transit, and while in use, which removes Microsoft from the trust chain.
- Location policies, which are also deployed by default to provide cloud admin control over where Azure resources can be deployed.
- Key management, which is controlled by a Federal Information Processing Standard (FIPS) 140-2 level-3 validated HSM and enforced by policy.

The policies and opinions that the sovereign landing zone (preview) adds on top of the Azure landing zone create a platform that's biased toward increased security and confidentiality by default.

For more information about the sovereignty policy baseline initiative, review the [Microsoft Cloud for Sovereignty policy portfolio](#) documentation.

Network topology and connectivity

The sovereign landing zone (preview) focuses on operational control of data at rest, in transit, and in use.

Network traffic encryption

For best practices for network encryption, see [Define network encryption requirements](#).

Internet inbound and outbound connectivity

Similar to Azure landing zone deployments, the sovereign landing zone deployment supports:

- A parameterized deployment of the premium tier of Azure Firewall, for enabling distributed denial-of-service (DDoS) protection.
- The deployment of a central Azure Bastion infrastructure.

Before you turn on these features, consult the best practices for internet inbound and outbound connectivity in [Plan for inbound and outbound internet connectivity](#).

Security

The sovereign landing zone architecture makes use of confidential computing in the confidential landing zones. The following sections describe services that provide support for Azure confidential computing.

Azure Key Vault Managed HSM

Key Vault is a necessary service for deploying confidential computing resources. For design considerations and recommendations, see [Encryption and key management in Azure](#). You might need to choose Azure Key Vault Managed HSM for compliancy requirements.

Azure Attestation

If you use Azure confidential computing, you can take advantage of the guest attestation feature of Azure Attestation. This feature helps to confirm that a confidential

VM runs on a hardware-based trusted execution environment (TEE) with security features like isolation and integrity enabled.

For more information about enabling guest attestation, see [What is guest attestation for confidential VMs?](#).

Governance

In most cases, Microsoft personnel perform operations, support, and troubleshooting, and no access to customer data is required. Occasionally, a Microsoft engineer needs to access customer data. These cases can come up in response to customer-initiated support tickets or when Microsoft identifies a problem.

Customer Lockbox for Microsoft Azure

In the rare circumstances when access is required, you can use [Customer Lockbox for Microsoft Azure](#). This feature provides an interface that you can use to review and then approve or reject customer data access requests.

Platform automation and DevOps

The sovereign landing zone (preview) is available as a [GitHub repository](#).

Deployment options

You can deploy the entire landing zone, or you can deploy one component at a time. When you deploy individual components, you can integrate them into your existing deployment workflow. For deployment guidance, see [Key components of the sovereign landing zone preview deployment](#).

Note

The sovereign landing zone (preview) is a variant of the Azure landing zone. But the sovereign landing zone doesn't yet offer all the deployment choices that are available for the Azure landing zone architecture. For information about deploying a sovereign landing zone, see [Key components of the sovereign landing zone preview deployment](#).

The GitHub repository includes the following sovereign landing zone (preview) components:

- Bootstrap: Sets up the management group hierarchy and creates the subscriptions as dictated by the architecture of the sovereign landing zone (preview). These elements are deployed under the tenant root group of the Azure customer tenant.
- Platform: Sets up the hub network and the logging resources that are used by the sovereign landing zone (preview) platform and workloads.
- Compliance: Creates and assigns the default policy sets and the custom policies that are enforced in the environment.
- Dashboard: Provides you with a visual representation of your resource compliance.

Compliance dashboard

A compliance dashboard is deployed as part of the sovereign landing zone (preview) deployment. This dashboard helps you validate the sovereign landing zone (preview) against your requirements and local laws and regulations. Specifically, the dashboard gives you insight into resource-level compliance against:

- The baseline policies that are deployed with the sovereign landing zone (preview).
- Other custom compliance that has been deployed.

For more information, see the [Compliance dashboard documentation](#).

Feedback

Was this page helpful?



How to get help building a landing zone

Article • 12/12/2024

Getting your Azure landing zone (ALZ) done right and on time is important. Working with a certified Azure partner is a great way to get the support you need to build your ALZ.

We'll show you how to find a partner and what to expect from your partner.

There are two options to find an ALZ partner (*see image*).

- *Option 1* - use Azure Migrate and Modernize.
- *Option 2* - find a partner offer for a landing zone in our marketplace.

Option 1: Azure Migrate and Modernize

What to expect:

1. ALZ planning
2. ALZ building
3. Deployment planning

Option 2: Partner Offer

What to expect:

1. Find offers
2. Meet potential partners
3. Understand the offer
4. Pick an offer

Option 1 - Azure Migrate and Modernize

The easiest option to find an ALZ partner is through Azure Migrate and Modernize.

How to find an Azure Migrate and Modernize partner

Take 5 minutes to [fill out the Azure Migrate and Modernize form ↗](#). Someone from the team will help you find the right partner.

What to expect from an Azure Migrate and Modernize partner

All the partners in the program adhere to ALZ best-practices. We vet partners to ensure customers receive ALZ guidance that follows CAF principles. Here are some of the

deliverables you and your partner will work through together.

1. *ALZ planning* – The Azure Migrate and Modernize partner will work with you to create the business and technical foundation needed to build your ALZ. ALZ planning includes:

- Selecting your operating models
- Reviewing your ALZ implementation options
- Identifying your compliance requirements
- Identifying any customization needed

2. *ALZ building* – Your partner will build and monitor an ALZ pilot. ALZ building includes:

- Reviewing your ALZ objectives and outcomes
- Defining your ALZ build plan
- Building your ALZ pilot
- Monitoring your ALZ pilot

3. *Deployment planning* – Your partner will work with you to define your deployment plan. Deployment planning includes:

- Assessing existing workload compatibility
- Presenting deployment plan options
- Defining resources, timeline, and next steps to execute the deployment plan

Option 2 - Partner offers

We have a self-discovery process that you can use to find a partner. The [Find an Azure partner](#) page provides information about working with certified Azure partners and links to the offers.

Offers come from Azure advanced specialization partners and Azure Expert Managed Services Providers (MSPs). Microsoft vets and approves each offer before it enters Azure Marketplace.

Advanced specialization partners demonstrate deep technical knowledge, proven success, and ethical business practices. Azure Expert MSPs also meet these criteria but must additionally pass a time- and cost-intensive auditing process.

Offers in the marketplace have a price, description, and contact information. Like any marketplace, there's flexibility in what the offer can be. Partners design and propose offers. Microsoft approves them for the marketplace.

This flexibility creates value for you and allows you to find an offer that meets your specific needs.

How to find partner offers

1. Go to the [Find an Azure partner](#) page. This page provides information about both Azure advanced specialization partners and Azure Expert MSPs.

In the **Azure advanced specialization partners** section, there's a list of links for specific types of migrations. Available offers are listed on the resulting pages.

In the **Azure Expert MSPs** section, select **Find an Azure Expert Managed Service Provider** to go to a similar list of available offers.

Each tile or square on these offer pages represents a different offer. Some partners have multiple offers, so you might see a partner appear more than once. Selecting a tile takes you to the partner offer. (There's also a button for contacting the partner. See the next step.)

- You can filter the results by expanding the categories in the **Filters** section on the left side of the page. You can use the search box to filter that list. You can also filter the results by location.
- Look for landing zone offers. Not every tile on the page will be a landing zone offer. Select **Contact me** to fill out a contact form. (See the next step.) Select anywhere else in the tile to see the details about the partner's offerings. The partner's technology specializations are listed in the tile. If there are more than one, there's a blue number next to the technology. Hover over the blue number to see the full list.

2. *Meet potential partners* – Use the **Contact me** button to reach out to partners.

Meet with several landing zone partners to find the right fit. Get a sense of their process, experience, and successes.

3. *Understand the offers* – Partners have more freedom to customize their ALZ approach outside of Azure Migrate and Modernize. Read the offer. Ask follow-up questions. Make sure you understand the offer and define expectations.

4. *Pick an offer* - Pick the offer that best meets your needs. Formalize the agreement and start building your landing zone with your partner.

What to expect from partner offers

Partner offers are vetted and approved for the marketplace. They follow CAF principles in general but have more freedom to customize their approach to the ALZ process. Expectations should align with the offer description and any other agreements made between you and your chosen partner.

Links to find a landing zone partner

- Option 1 - [Azure Migrate and Modernize](#)
- Option 2 - [Find an Azure partner](#)

Next steps

Learn about the process for refactoring landing zones.

[Refactor landing zones](#)

Feedback

Was this page helpful?

 Yes

 No

Transition an existing Azure environment to the Azure landing zone conceptual architecture

Article • 04/01/2024

Many organizations have an existing Azure footprint, one or more subscriptions, and potentially an existing management group structure. Depending on their business requirements and scenarios, they might have Azure resources deployed, such as Azure VPN Gateway or Azure ExpressRoute for hybrid connectivity.

This article provides recommendations to help your organization navigate changes based on your existing Azure environment that's transitioning into the Azure landing zone conceptual architecture. This article also describes considerations for moving resources in Azure, for example moving a subscription from one existing management group to another management group. Consider these recommendations to help you evaluate and plan the transition of your existing Azure environment.

Move resources in Azure

You can move some resources in Azure after creation. There are different approaches that are subject to a user's Azure role-based access control (RBAC) permissions at and across scopes. The following table outlines which resources you can move, at which scope, and the pros and cons associated with each resource.

[\[\] Expand table](#)

Scope	Destination	Pro	Con
Resources in resource groups.	You can move to a new resource group in the same or different subscription.	You can modify the resource composition in a resource group after deployment.	Not supported by all resourceTypes. Some resourceTypes have specific limitations or requirements. resourceIds are updated and it affects existing monitoring, alerts, and control plane operations. Resource groups are

Scope	Destination	Pro	Con
			locked during the move period.
Subscriptions in a tenant.	You can move to different management groups.	No effect on existing resources within the subscription because resourceId values don't change.	Requires an assessment of policies and RBAC pre-move and post-move operation.

To determine which move strategy you should use, consider the following examples.

Move subscriptions

Typically, you move subscriptions to organize them into management groups or to transfer subscriptions to a new Microsoft Entra ID tenant. Moving a subscription to a new tenant is mainly for [transferring billing ownership](#). For more information about how to move subscriptions across management groups in the same tenant, see [Moving management groups and subscriptions](#).

Azure RBAC requirements

To assess a subscription prior to a move, it's important that the user has the appropriate Azure RBAC. The user might be an owner on the subscription (direct role assignment) and have write permission on the target management group. Built-in roles that support write permission on the target management group are the owner role, the contributor role, and the management group contributor role.

If the user has an inherited owner role permission on the subscription from an existing management group, you can only move the subscription to the management group in which the user is assigned the owner role.

Policies

Existing subscriptions might be subject to Azure policies that are assigned directly or assigned at the management group where they're currently located. It's important to

assess current policies and the policies that might exist in the new management group or the management group hierarchy.

You can use Azure Resource Graph to perform an inventory of existing resources and compare their configuration with the policies that exist at the destination.

After you move subscriptions to a management group with existing Azure RBAC and policies in place, consider the following factors:

- For any Azure RBAC that's inherited to the moved subscriptions, the user tokens in the management group cache might take up to 30 minutes to refresh. To expedite this process, you can refresh the token by signing out and in, or request a new token.
- A policy in which the assignment scope includes the moved subscriptions performs an audit only on the existing resources. An existing resource in the subscription that's subject to a:
 - `DeployIfNotExists` policy effect appears as noncompliant and isn't automatically remediated. A user must manually perform the remediation.
 - `Deny` policy effect appears as noncompliant and isn't rejected. A user must manually mitigate this result as appropriate.
 - `Append` and `Modify` policy effect appears as noncompliant and requires a user to mitigate.
 - `Audit` and `AuditIfNotExist` policy effect appears as noncompliant and requires a user to mitigate.
- All new writes to resources in the moved subscription are subject to the assigned policies in real time like normal.

Move resources

Typically, you move resources when you want to consolidate resources into the same resource group if they share the same lifecycle. Or if you want to move resources to a different subscription due to cost, ownership, or Azure RBAC requirements.

When you move resources, the source resource group and the target resource group are locked during the move operation. You can't add, update, or delete resources in the resource groups. A resource move operation doesn't change the location of the resources.

For more information about how to move resources across resource groups and subscriptions in the same tenant, see [Move resources to a new resource group or subscription](#).

💡 Tip

To minimize the effect of regional outages, we recommend that you place resources in the same region as the resource group. For more information, see [Resource group location alignment](#).

If you have resources in different regions within the same resource group, consider moving your resources to a [new resource group or subscription](#).

To [determine if your resource supports moving to another resource group](#), inventory your resources by cross-referencing them. Ensure that you meet the appropriate [prerequisites](#).

Before you move resources

Prior to a move operation, you must verify that the [resources are supported](#), and assess their requirements and dependencies. For example, when you move a peered virtual network, you have to disable virtual network peering first, and re-enable peering after the move operation finishes. Plan in advance for the disable and re-enable dependency so you understand the effect on existing workloads that might be connected to your virtual networks.

After you move resources

When you move the resources into a new resource group in the same subscription, any inherited Azure RBAC and policies from the management group or subscription still apply. This also applies if you move to a resource group in a new subscription where the subscription might be subject to other Azure RBAC and policy assignment. You have to validate the resource compliance and access controls.

Scenarios

The following scenarios describe how to migrate and transition an existing environment into the Azure landing zone conceptual architecture.

- Alignment scenarios

- Transition a single subscription with no management groups to the Azure landing zone conceptual architecture
 - Transition management groups to the Azure landing zone conceptual architecture
 - Transition a regional organization to the Azure landing zone conceptual architecture
- **Alignment approaches**
 - Transition an environment by duplicating a landing zone management group

Journey toward the target architecture

Feedback

Was this page helpful?

 Yes

 No

Scenario: Transition a single subscription with no management groups to the Azure landing zone conceptual architecture

Article • 12/16/2024

This article describes considerations and instructions to migrate and transition your Azure environment into the Azure landing zone conceptual architecture. This scenario covers a single subscription with no management groups.

In this scenario, the customer already uses Azure and already hosts a few applications or services within the platform. But their current implementation limits their scalability and growth related to their *cloud first* strategy.

As part of this expansion, they plan to migrate away from their on-premises datacenters and into Azure. During the migration, they lead with modernizing and transforming their applications or services to use cloud-native technologies where possible. For example, they might use Azure SQL Database and Azure Kubernetes Service (AKS). They know that it takes considerable time and effort, so they plan to *lift and shift* to start. Initially, this plan requires hybrid connectivity via services such as Azure VPN Gateway or Azure ExpressRoute.

The customer wants to move from their existing approach to an enterprise-scale architecture. This architecture supports their *cloud first* strategy and has a robust platform that scales as the customer eliminates their on-premises datacenters.

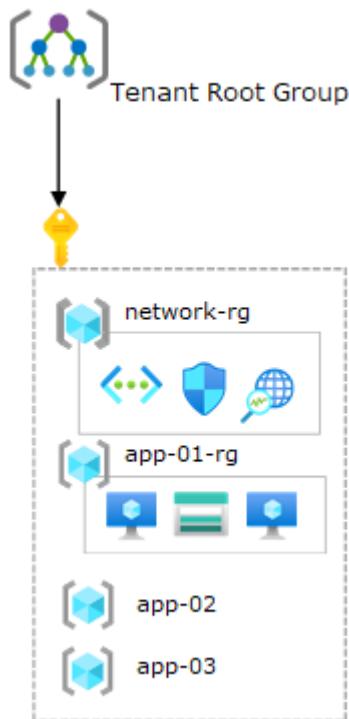
Current state

In this scenario, the current state of the customer's Azure environment consists of:

- A single Azure subscription.
- No custom management groups.
- Nonuniform resource distribution. Platform and workload resources are deployed in the same Azure subscription.
- Minimal usage of Azure Policy. Policy assignments, such as audit effects and deny effects, are performed for each resource group, with exceptions.
- Resource groups that are treated as units of management and scale.
- Role-based access control role assignments for each resource group.
- A single virtual network.

- No hybrid connectivity via services such as Azure VPN Gateway or Azure ExpressRoute.
- A new subnet is created for each application.
- Multiple self-contained applications in each of the *app-xx-rg* resource groups. Applications are controlled and operated by different application or service teams.

The following diagram shows the current state of this scenario.



Transition to the Azure landing zone conceptual architecture

Prior to implementing this approach, review [Azure landing zone conceptual architecture](#) and [Azure landing zone design areas](#).

To transition from this scenario's current state to an Azure landing zone conceptual architecture, use this approach:

1. Deploy the [Azure landing zone accelerator](#) into the same Microsoft Entra ID tenant in parallel with the current environment. This method provides a smooth and phased transition to the new landing zone architecture with minimal disruption to active workloads.

This deployment creates a new management group structure. This structure aligns with Azure landing zones design principles and recommendations. It also ensures that these changes don't affect the existing environment.

2. (Optional) Work with application or service teams to migrate the workloads that are deployed in the original subscription into new Azure subscriptions. For more information, see [Transition existing Azure environments to the Azure landing zone conceptual architecture](#). You can place workloads into the newly deployed Azure landing zone conceptual architecture management group hierarchy under the correct management group, such as *corporate* or *online* in the following diagram.

For details about the effect on resources when migrating, see [Policies](#).

Eventually, you can cancel the existing Azure subscription, and place it in the decommissioned management group.

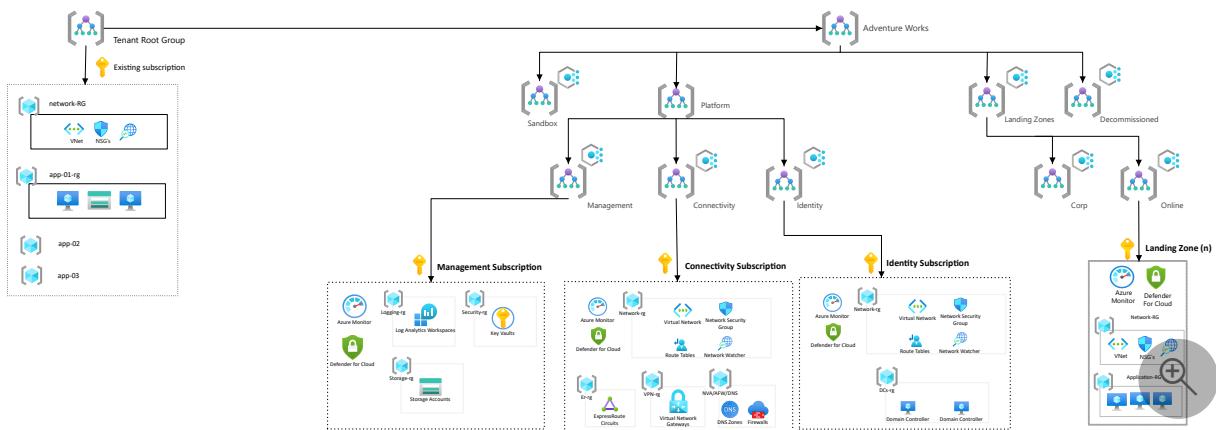
ⓘ Note

You don't necessarily have to migrate the existing applications or services into new landing zones, or Azure subscriptions.

3. Create new Azure subscriptions to provide landing zones that can support migration projects from on-premises. Place them under the proper management group, such as *corporate* or *online* in the following diagram.

For more information, see [Readying your landing zone for migration guidance](#).

The following diagram shows the state of this scenario during the migration.



Summary

In this scenario, the customer accomplished their expansion and scaling plans within Azure by deploying the [Azure landing zone conceptual architecture](#) in parallel to their existing environment.

Feedback

Was this page helpful?

 Yes

 No

Scenario: Transition management groups to the Azure landing zone conceptual architecture

Article • 10/16/2023

This article describes considerations and instructions to migrate and transition your existing environment into the conceptual architecture of Azure landing zones. This scenario covers single or multiple management groups.

In this scenario, it's assumed that the customer already uses Azure. They have a management group hierarchy with multiple subscriptions that host a few applications or services within the platform. But their current implementation limits their scalability and growth related to their *cloud first* strategy.

As part of this expansion, they plan to migrate away from their on-premises datacenters and into Azure. During the migration, they lead with modernizing and transforming their applications or services to use cloud-native technologies where possible. For example, they might use Azure SQL Database and Azure Kubernetes Service (AKS). They know that it takes considerable time and effort, so they plan to *lift and shift* to start. Initially, this plan requires hybrid connectivity via services such as Azure VPN Gateway or Azure ExpressRoute.

The customer wants to move their existing environment to the Azure landing zones conceptual architecture. This architecture supports their *cloud first* strategy and has a robust platform that scales as the customer retires their on-premises datacenters.

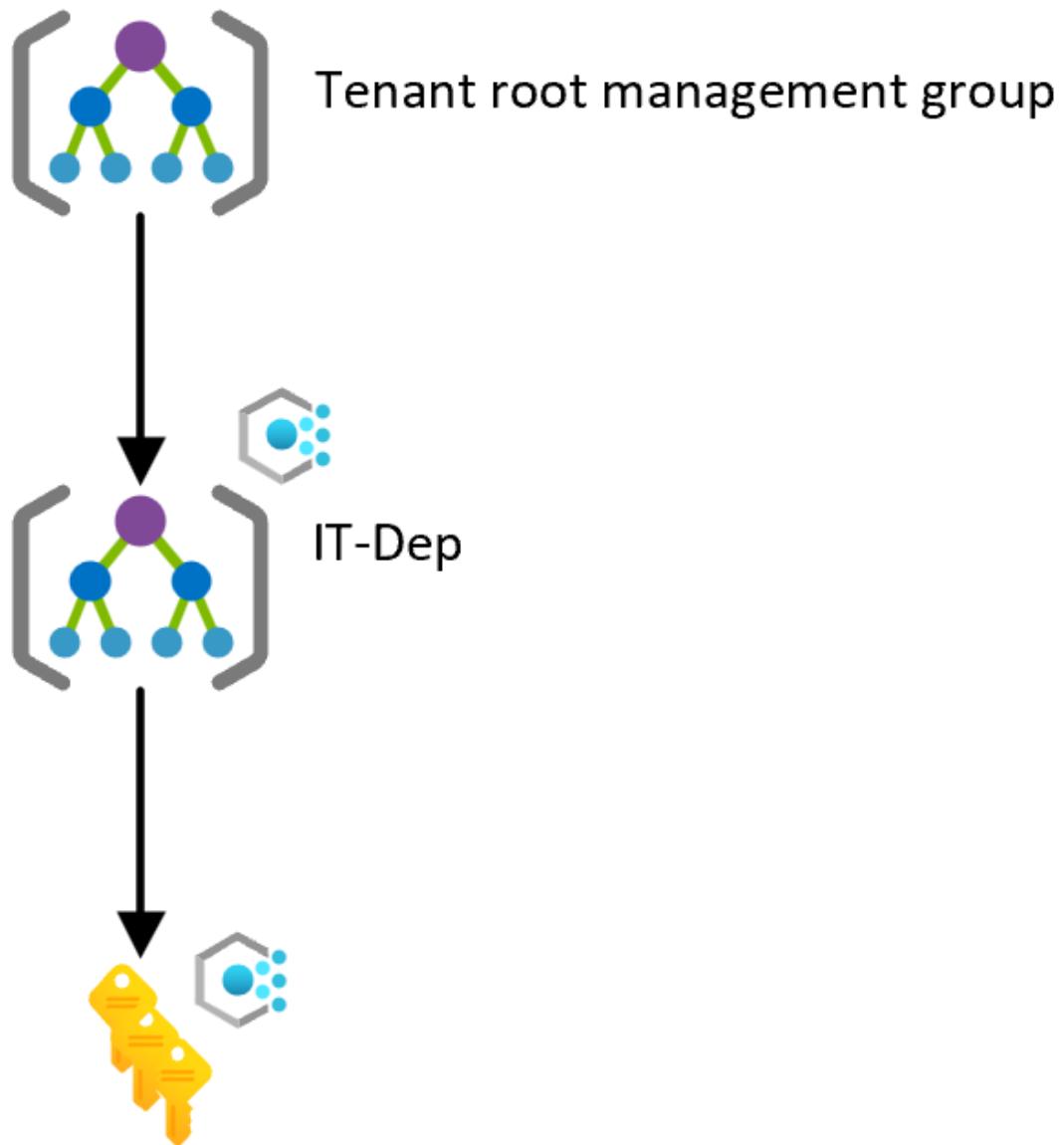
Current state

In this scenario, the current state of the customer's Azure environment consists of:

- One or more management groups.
- A management group hierarchy that's based on organizational structure or geography.
- An Azure subscription for each application environment, like development, testing, or production (dev/test/prod).
- Nonuniform resource distribution. Platform and workload resources for a single environment are deployed in the same Azure subscriptions.
- Policy assignments with audit effects and deny effects that are assigned at the management group and subscription level.

- Role-based access control role assignments for each subscription and resource group.
- A hub virtual network, such as Azure VPN Gateway or Azure ExpressRoute, for hybrid connectivity.
- A virtual network for each application environment.
- Applications that are deployed into the respective subscription based on their environment classification, such as development, testing, or production.
- A central IT team that controls and operates the environment.

The following diagram shows the current state of this scenario.



Transition to the Azure landing zone conceptual architecture

Prior to implementing this approach, review [Azure landing zone conceptual architecture](#), [Azure landing zone design principles](#), and [Azure landing zone design areas](#).

To transition from this scenario's current state to an Azure landing zone conceptual architecture, use the following approach:

1. Deploy the [Azure landing zone accelerator](#) into the same Microsoft Entra ID tenant in parallel with the current environment. This method provides a smooth and phased transition to the new landing zone architecture with minimal disruption to active workloads.

This deployment creates a new management group structure. This structure aligns with Azure landing zones design principles and recommendations. It also ensures that these changes don't affect the existing environment.

For more information, see [How to handle a dev/test/prod workload landing zone](#).

2. (Optional) Work with application or service teams to migrate the workloads that are deployed in the original subscriptions into new Azure subscriptions. For more information, see [Transition an existing Azure environment to the Azure landing zone conceptual architecture](#). You can place workloads into the newly deployed Azure landing zone conceptual architecture management group hierarchy under the correct management group, such as *corporate* or *online* in the following diagram.

For details about the effect on resources when migrating, see [Policies](#).

Eventually, you can cancel the existing Azure subscription, and place it in the decommissioned management group.

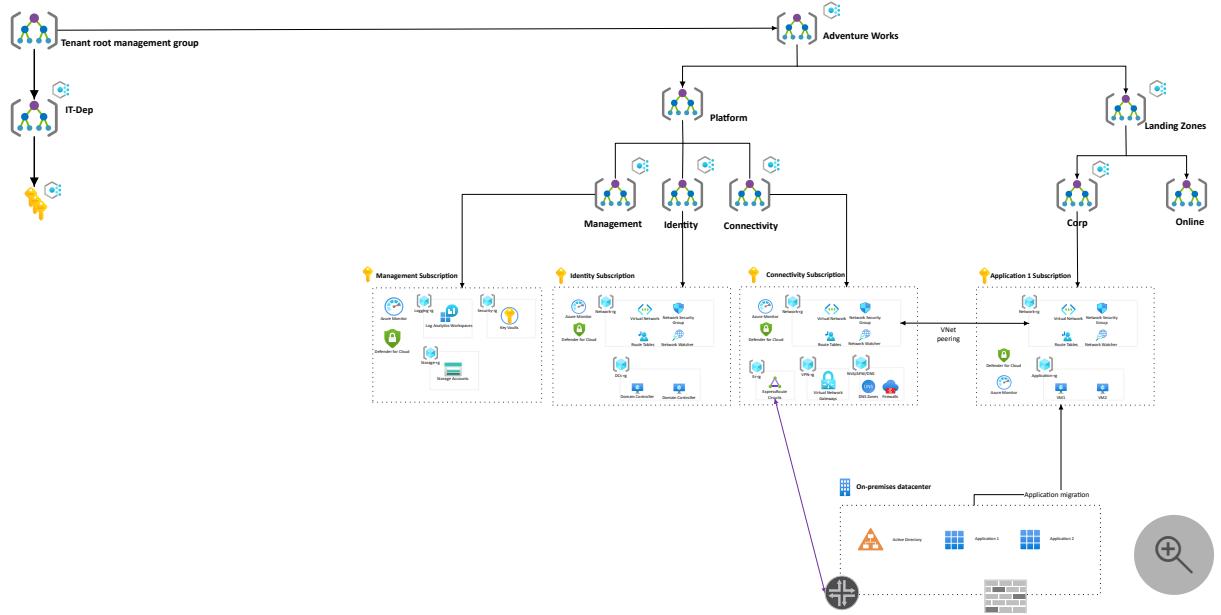
 **Note**

You don't necessarily have to migrate the existing applications or services into new landing zones, or Azure subscriptions.

3. Create new Azure subscriptions to provide landing zones that support migration projects from on-premises. Place them under the proper management group, such as *corporate* or *online* in the following diagram.

For more information, see [Readying your landing zone for migration](#).

The following diagram shows the state of this scenario during the migration.



Summary

In this scenario, the customer accomplished their expansion and scaling plans within Azure by deploying the [Azure landing zone conceptual architecture](#) parallel to their existing environment.

Scenario: Transition a regional organization environment to the Azure landing zone conceptual architecture

Article • 10/16/2023

This article describes considerations and instructions to migrate and transition your Azure environment into the Azure landing zone conceptual architecture. This scenario covers a regional organization with management groups that are separated into development, testing, and production (dev/test/prod) environments.

In this scenario, the customer has a large footprint on Azure. They have a management group hierarchy that's organized by dev/test/prod environments and then by region. Their current implementation limits their scalability and growth. They have applications deployed across the globe. A central IT team manages each region. In this scenario the regions are America; Europe, the Middle East, and Africa (EMEA); and Asia-Pacific (APAC).

The customer wants to move from their existing environment to the Azure landing zones conceptual architecture. This approach supports their *cloud first* strategy and has a robust platform that scales as the customer retires their on-premises datacenters.

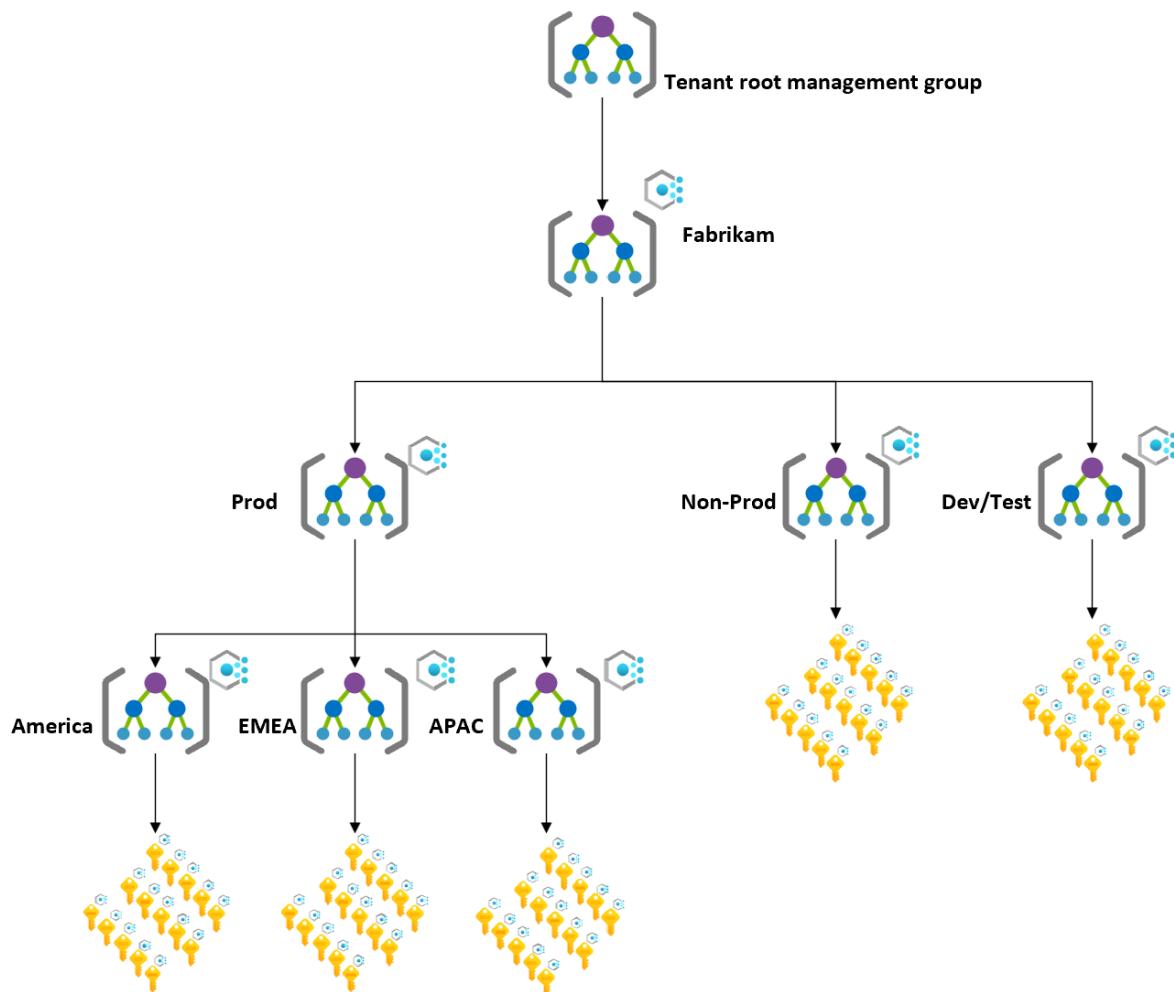
Current state

In this scenario, the current state of the customer's Azure environment consists of:

- Multiple management groups.
- A management group hierarchy based on dev/test/prod environments at the first level and then based on geography at the second level.
- An Azure subscription for each geography and application environment, such as dev/test/prod. This subscription is required to provide developers with a relaxed environment for testing and innovation.
- Some critical workloads that need the same governance model across dev/test/prod, which can create governance challenges for the customer.
- Nonuniform resource distribution. Platform and workload resources for a single environment are deployed in the same Azure subscriptions.
- Applications that are deployed into the respective subscriptions based on their region and environment classification, like dev/test/prod.
- Policy assignments, such as audit effects and deny effects, that are assigned at the management group and subscription level.

- The same set of Azure policies applied to all applications in the same region and in the same environment type.
- Role-based access control role assignments for each subscription and resource group.
- A hub virtual network, such as Azure VPN Gateway or Azure ExpressRoute, for hybrid connectivity.
- A virtual network for each application environment.
- A central IT team that controls and operates the respective management group for each region. The team faces some consistency, configuration, and compliance challenges when it comes to policies, access control, platform resources configuration, and security compliance because some applications are deployed into multiple regions.

The following diagram shows the current state of this sample scenario.



Transition to the Azure landing zone conceptual architecture

Prior to implementing this approach, review [Azure landing zone conceptual architecture](#), [Azure landing zone design principles](#), and [Azure landing zone design areas](#).

To transition from this scenario's current state to an Azure landing zone conceptual architecture, use this approach:

1. Deploy the [Azure landing zone accelerator](#) into the same Microsoft Entra ID tenant in parallel with the current environment. This method provides a smooth and phased transition to the new landing zone architecture with minimal disruption to active workloads.

This deployment creates a new management group structure. This structure aligns with Azure landing zones design principles and recommendations. It also ensures that these changes don't affect the existing environment.

For more information, see [How to handle a dev/test/prod workload landing zone](#).

For information about using sandbox management group hierarchy to empower developers to test and experiment without affecting other environments, see [Azure Landing zone sandbox environments guidance](#).

For information about minimizing disruption to applications and services during the migration, see [Adopt policy-driven guardrails guidance](#).

2. (Optional) Work with application or service teams to migrate the workloads that are deployed in the original subscriptions into new Azure subscriptions. For more information, see [Transition existing Azure environments to the Azure landing zone conceptual architecture](#). You can place workloads into the newly deployed Azure landing zone conceptual architecture management group hierarchy under the correct management group, such as *corporate* or *online* in the following diagram.

For details about the effect on resources when migrating, see [Policies](#).

Eventually, you can cancel the existing Azure subscription, and place it in the decommissioned management group.

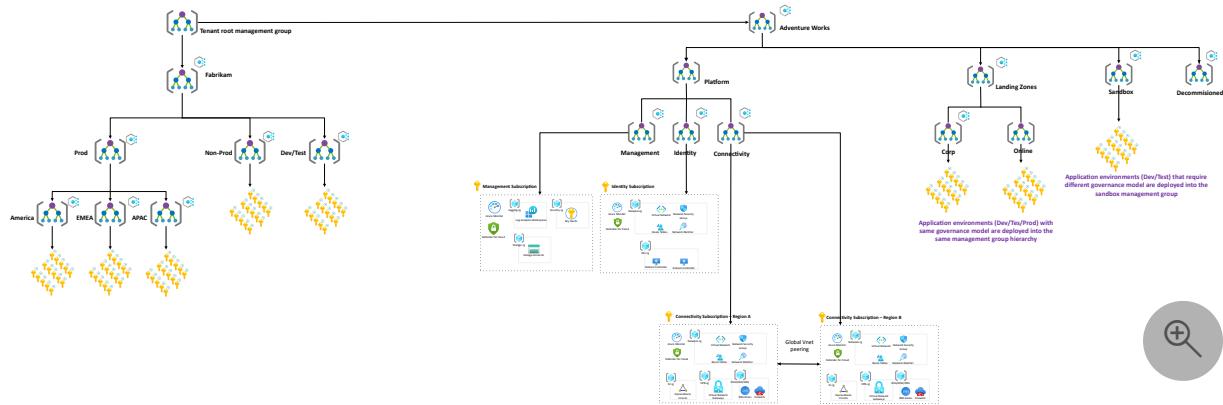
Note

You don't necessarily have to migrate the existing applications or services into new landing zones, or Azure subscriptions.

3. Create new Azure subscriptions to provide landing zones that can support new applications and workloads. Place them under the proper management group, such as *corporate* or *online* in the following diagram.

For more information, see [Readyng your landing zone for migration guidance](#).

The following diagram shows the state of this scenario during the migration.



Summary

In this scenario, the customer established the necessary foundation to support their growth and scale plans for their workloads in Azure by deploying the [Azure landing zone conceptual architecture](#) in parallel to their existing environment.

Scenario: Transition an environment by duplicating a landing zone management group

Article • 10/16/2023

This article describes an example approach that transitions an environment to the Azure landing zone conceptual architecture by duplicating the landing zone management group with policies in *audit only* mode. With this approach, you can quickly access the new desired target architecture and then assess the application or workload subscriptions for compliance. This approach eliminates the risk of affecting the application teams because the policies are in *audit only* mode.

Transition to the Azure landing zone conceptual architecture

Prior to implementing this approach, review [Azure landing zone conceptual architecture](#), [Azure landing zone design principles](#), and [Azure landing zone design areas](#).

Use this approach to transition to the Azure landing zone conceptual architecture:

1. Deploy the [Azure landing zone accelerator](#) into the same Microsoft Entra ID tenant in parallel with the current environment. This method provides a smooth and phased transition to the new landing zone architecture with minimal disruption to active workloads.

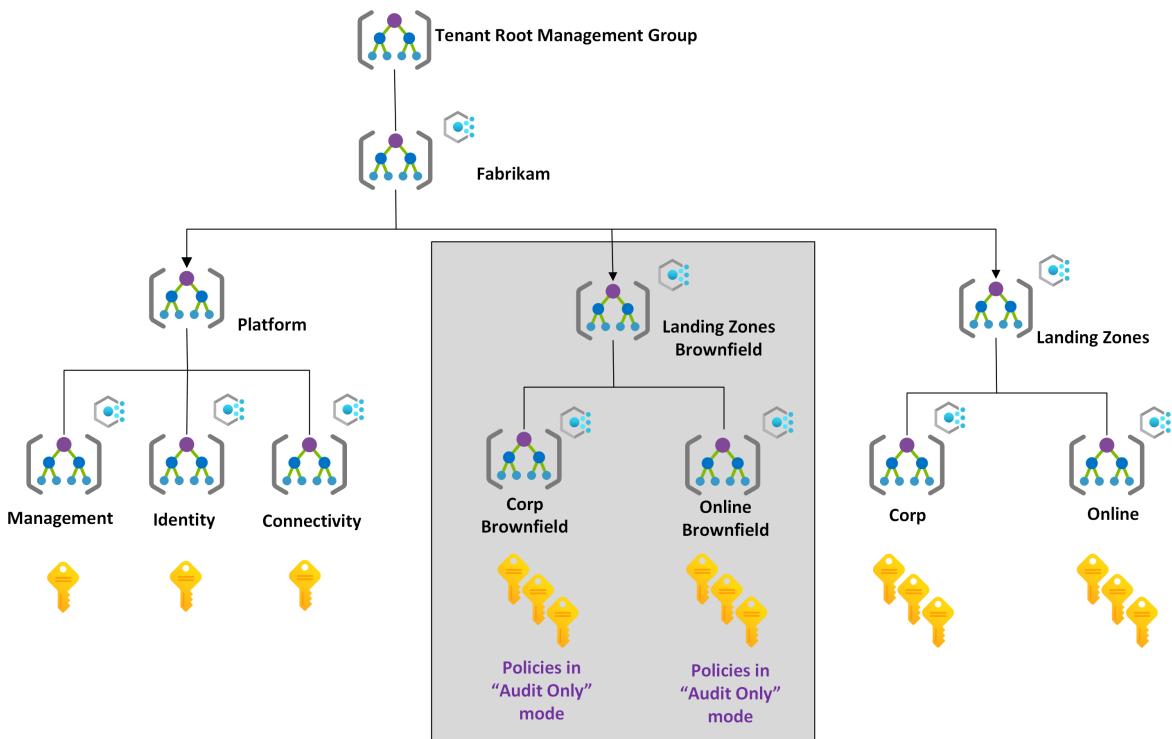
This deployment creates a new management group structure. This structure aligns with Azure landing zone design principles and recommendations. It also ensures that these changes don't affect the existing environment.

For information about minimizing disruption to applications and services during the migration, see [Adopt policy-driven guardrails](#).

2. To duplicate the landing zone management group and its children (*corp* and *online* in the following diagram) including all policy assignments, configure them to *audit only* mode. On the policy assignments, set the [enforcementMode property](#) to `DoNotEnforce` or `Disabled`.

This approach provides quick access to the new desired target architecture. Then the applications teams can assess the policies without the risk of affecting active

applications.



➊ Note

This approach has no additional cost because it only duplicates the management group hierarchy and the assigned policies, not the workloads.

3. (Optional) Work with application or service teams to migrate the workloads that are deployed in the original subscriptions into new Azure subscriptions. For more information, see [Transition existing Azure environments to the Azure landing zone conceptual architecture](#). You can place workloads into the newly duplicated management group hierarchy under the correct management group, such as *corporate brownfield* or *online brownfield* in this example.

For information about the effect on resources when migrating, see [Policies](#).

Eventually, you can cancel the existing Azure subscription, and place it in the decommissioned management group.

➋ Note

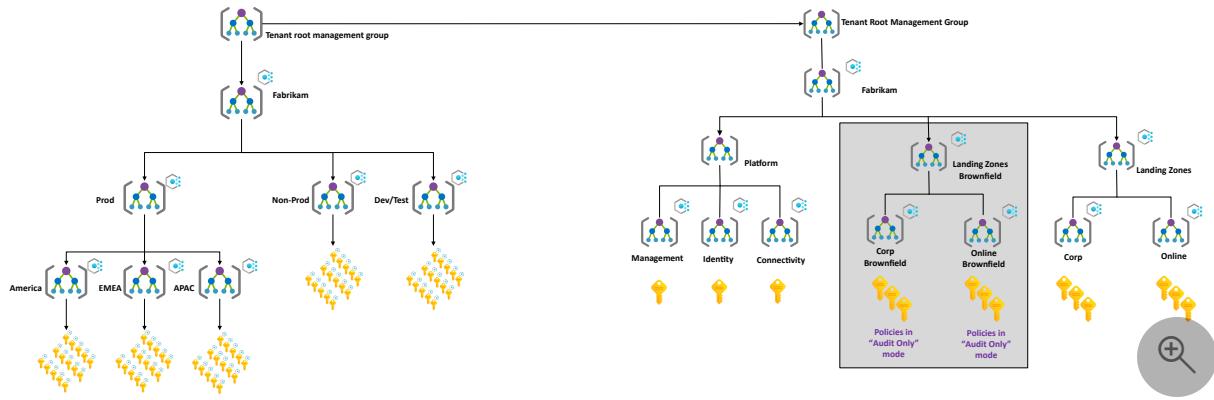
You don't necessarily have to migrate the existing applications or services into new landing zones, or Azure subscriptions.

4. After the application teams work with the platform teams to get their policy compliance into the required state, their subscriptions are moved to the proper

management group, such as *corporate* or *online* in the following diagram. They're covered by the assigned policies and your team can efficiently and compliantly operate their workload.

For more information, see [Readying your landing zone for migration guidance](#).

The following diagram shows the state of this scenario during the migration.



Summary

You used this approach to safely migrate your workloads in Azure by deploying the [Azure landing zone conceptual architecture](#) in parallel with your existing environment with minimal disruption.

Brownfield landing zone considerations

Article • 11/18/2024

A brownfield deployment is an existing environment that requires modification to align to the Azure landing zone target architecture and best practices. When you need to resolve a brownfield deployment scenario, consider your existing Microsoft Azure environment as the place to start. This article summarizes guidance found elsewhere in the Cloud Adoption Framework Ready documentation. For more information, see [Introduction to the Cloud Adoption Framework Ready methodology](#).

Resource organization

In a brownfield environment, you've already established your Azure environment. But it's never too late to apply proven [resource organization principles](#) now and moving forward. Consider implementing any of the following suggestions:

- If your current environment doesn't use management groups, consider them. Management groups are key to managing policies, access, and compliance across subscriptions at scale. [Management groups](#) help guide your implementation.
- If your current environment uses management groups, consider the guidance in [management groups](#) when evaluating your implementation.
- If you have existing subscriptions in your current environment, consider the guidance in [subscriptions](#) to see if you're using them effectively. Subscriptions act as policy and management boundaries and are scale units.
- If you have existing resources in your current environment, consider using the guidance in [naming and tagging](#) to influence your tagging strategy and your naming conventions going forward.
- [Azure Policy](#) is useful in establishing and enforcing consistency regarding taxonomic tags.

Security

To refine your existing Azure environment's [security posture](#) regarding authentication, authorization, and accounting is an ongoing, iterative process. Consider implementing the following recommendations:

- Deploy [Microsoft Entra Connect cloud sync](#) to provide your local Active Directory Domain Services (AD DS) users with secure single sign-on (SSO) to your Microsoft Entra ID-backed applications. Another benefit to configuring hybrid identity is you

can enforce [Microsoft Entra multifactor authentication \(MFA\)](#) and [Microsoft Entra Password Protection](#) to further protect these identities

- Provide secure authentication to your cloud apps and Azure resources by using [Microsoft Entra Conditional Access](#).
- Implement [Microsoft Entra Privileged Identity Management](#) to ensure least-privilege access and deep reporting in your entire Azure environment. Teams should begin recurring access reviews to ensure the right people and service principles have current and correct authorization levels. Also, study the [access control guidance](#).
- Make use of the recommendations, alerting, and remediation capabilities of [Microsoft Defender for Cloud](#). Your security team can also integrate Microsoft Defender for Cloud into [Microsoft Sentinel](#) if they need a more robust, centrally managed hybrid and multicloud Security Information Event Management (SIEM)/Security Orchestration and Response (SOAR) solution.

Governance

Like Azure security, [Azure governance](#) isn't a "one and done" proposition. Rather, it's an ever-evolving process of standardization and compliance enforcement. Consider implementing the following controls:

- Review our guidance for establishing a [management baseline](#) for your hybrid or multicloud environment
- Implement [Microsoft Cost Management](#) features like billing scopes, budgets, and alerts to ensure your Azure spend stays within prescribed bounds
- Use [Azure Policy](#) to enforce governance guardrails on Azure deployments, and trigger remediation tasks to bring existing Azure resources into a compliant state
- Consider [Microsoft Entra entitlement management](#) to automate Azure requests, access assignments, reviews, and expiration
- Apply [Azure Advisor](#) recommendations to ensure cost optimization and operational excellence in Azure, both of which are core principles of the [Microsoft Azure Well-Architected Framework](#).

Networking

It's true that refactoring an already established [Azure virtual network \(VNet\)](#) [infrastructure](#) can be a heavy lift for many businesses. That said, consider incorporating the following guidance into your network design, implementation, and maintenance efforts:

- Review our best practices for planning, deploying, and maintaining [Azure VNet hub and spoke topologies](#)
- Consider [Azure Virtual Network Manager \(Preview\)](#) to centralize network security group (NSG) security rules across multiple VNets
- [Azure Virtual WAN](#) unifies networking, security, and routing to help businesses build hybrid cloud architectures safer and quicker
- Access Azure data services privately with [Azure Private Link](#). The Private Link service ensures your users and applications communicate with key Azure services by using the Azure backbone network and private IP addresses instead of over the public Internet

Next steps

Now that you have an overview of Azure brownfield environment considerations, here are some related resources to review:

- [Azure Landing Zones Bicep - Deployment Flow](#)
- [Microsoft Well-Architected Framework](#)
- [Cloud Adoption Framework tools and templates](#)

Feedback

Was this page helpful?



Expand your landing zone

Article • 05/14/2024

A refactoring approach to infrastructure as code (IaC) removes blockers to business success and minimizes risk. This series of articles assumes that you deployed your first landing zone and want to expand that landing zone to meet enterprise requirements.

Shared architecture pillars

Landing zone expansion provides the opportunity to embed the following pillars into your landing zone and into your broader cloud environment.



These pillars are shared by [Azure Advisor](#), the [Microsoft Azure Well-Architected Framework](#), and the solutions in the [Azure Architecture Center](#).

Apply these pillars to landing zone improvements

To better align with the Cloud Adoption Framework, the pillars above are grouped into actionable landing zone improvements:

- Basic considerations: refactor a landing zone to refine hosting, fundamentals, and other foundational elements.
- Operations expansions: add operations management configurations to improve **performance, reliability, and operational excellence**.
- Governance expansions: add governance configurations to improve **cost, reliability, security**, and consistency.
- Security expansions: add **security** configurations to improve protection of sensitive data and critical systems.

Warning

If you have a midterm objective (within 24 months) to **host more than 1,000 applications, infrastructure assets, or data assets in the cloud**, consider these expansions early in the cloud adoption journey. For all other adoption patterns, landing zone expansions could be a parallel iteration.

Next step

Before refactoring your first landing zone, it's important to understand test-driven development.

[Test-driven development for landing zones](#)

Feedback

Was this page helpful?

 Yes

 No

Improve landing zone operations

Article • 02/24/2023

When you've achieved the [Ready](#) state and implemented [Azure landing zones](#), you still have an ongoing responsibility to manage your cloud environment in the most efficient way possible. This article provides guidance on improving landing zone operations as you scale, helping you meet growing operational excellence, reliability, and performance requirements.

The Manage methodology

The [Manage methodology](#) of the Cloud Adoption Framework provides guidance for establishing tooling for a management baseline and building operations management capacity across landing zones. It also outlines ways to extend your management baseline and build extra resiliency. We use the basic structure of the [Azure Management Guide](#) to improve operations across landing zones.

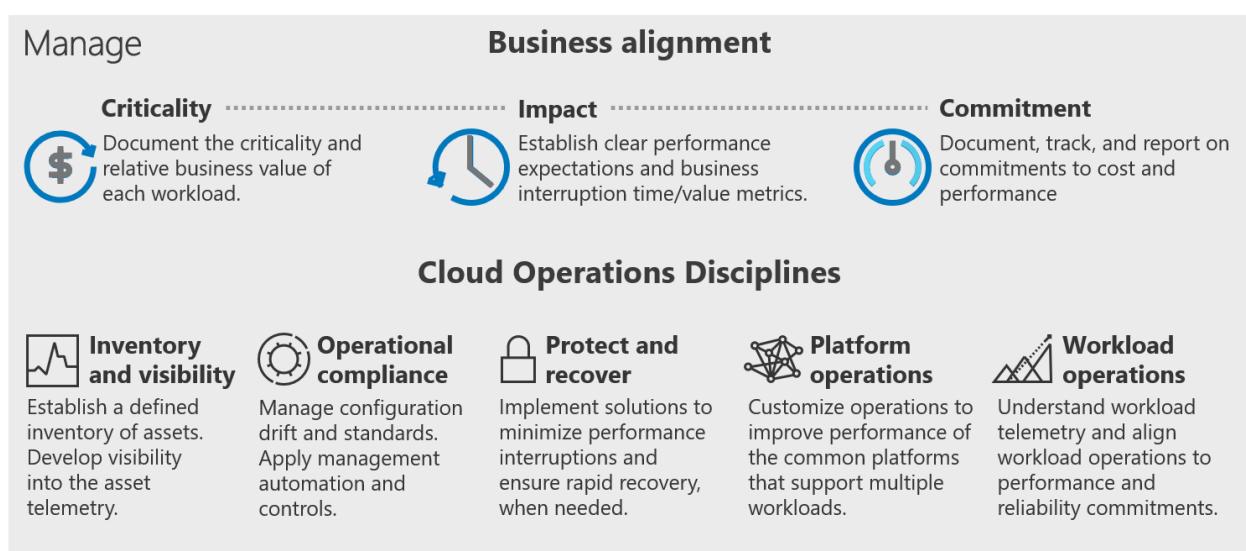


Figure 1: The Manage methodology of the Cloud Adoption Framework.

Prerequisite: Establish a management baseline

A management baseline establishes the foundation for operations management. It represents the minimum set of tools and processes that should be applied to every asset in a cloud environment. For high-level information, see [Cloud management disciplines](#) and [Overview of Azure server management services](#). The disciplines and guidance can be applied to any landing zone to improve initial operations.

An effective management baseline encompasses three areas:

1. Inventory and visibility

This discipline comes first because collecting proper operational data is vital when making decisions about operations. Cloud management teams must understand what is managed and how well those assets are operated. See [Inventory and visibility in Azure](#) and [Inventory and visibility in cloud management](#).

2. Operational compliance

Improving operational compliance reduces the likelihood of an outage related to configuration drift or vulnerabilities related to systems being improperly patched. See [Operational compliance in Azure](#) and [Operational compliance in cloud management](#).

3. Protect and recover

Protect and recover is the final discipline in any cloud-management baseline, aiming to reduce the duration and impact of outages that can't be prevented. See [Protect and recover in Azure](#) and [Protect and recover in cloud management](#).

Create business alignment

Once you've established a management baseline, you need to understand the criticality and impact of each workload within a landing zone in order to drive ongoing management improvements.

When an organization moves to the cloud, management and operations naturally shift a bit. This shift creates an opportunity to develop tighter business alignment by defining criticality, business impact, and business commitments. See [Create business alignment in cloud management](#).

Define criticality

Most businesses have a few workloads that are too important to fail, while other workloads can go months at a time without being used. Understanding the criticality of each workload in the IT portfolio is the first step toward establishing mutual commitments to cloud management. See [Business criticality in cloud management](#).

Understand business impact

In order to manage investments wisely, it's important to understand the impact on the business when an outage or performance degradation occurs. See [Business impact in cloud management](#).

Define business commitments

Defining business commitments is an exercise in balancing priorities. The objective is to align the proper level of operational management at an acceptable operating cost. See [Business commitment in cloud management](#).

Enhance the management baseline

Identify the workloads that require improving landing zone operations beyond the initial baseline. Some workloads might require enhancements to the baseline that aren't necessarily specific to a single platform or workload. Although these enhancements aren't cost effective for every workload, you should establish common processes, tools, and solutions for any workload that can justify the cost of the extra management support. See [Enhanced management baseline in Azure](#).

Apply advanced operations and design principles

Review the design and operations of specific workloads, platforms, or full landing zones to meet deeper requirements. In some cases, aspects of workload and platform operations might require changes to design and architecture principles. The two primary areas of management specialization are **platform specialization** and **workload specialization**. See [Apply design principles and advanced operations](#).

Platform specialization

Platform specialization means to invest in ongoing operations of a shared platform, distributing the investment across multiple workloads. See [Platform specialization for cloud management](#) and [Platform operations in cloud management](#).

Workload specialization

Workload specialization means to invest in ongoing operations of a specific workload, generally reserved for mission-critical workloads. See [Workload specialization for cloud management](#) and [Workload operations in cloud management](#).

Landing zone operations best practices

The following links provide best practices for improving landing zone operations.

- [Establish operational management practices in the cloud](#): This section of the Cloud Adoption Framework guides you through various transitions into operational management in the cloud.
- [Azure server management](#): An onboarding guide to incorporating the cloud-native tools and services needed to manage operations.
- [Hybrid monitoring](#): Many customers have already made a substantial investment in System Center Operations Manager. For those customers, this guide to hybrid monitoring can help them compare and contrast the cloud-native reporting tools with Operations Manager tooling. This comparison makes it easier to decide which tools to use for operational management.
- [Centralize management operations](#): Use Azure Lighthouse to centralize operations management across multiple Azure tenants.
- [Establish an operational fitness review](#): Review an environment for operational fitness.
- Workload specific operations best practices:
 - [Resiliency checklist](#)
 - [Failure mode analysis](#)
 - [Recover from a region wide service disruption](#)
 - [Recover from data corruption or accidental deletion](#)
- Build a Subscription Vending process to automate the creation of subscriptions for application teams via a request workflow as described in [Subscription vending](#)

Next steps

Understand how to [improve landing zone governance](#) to support adoption at scale.

[Improve landing zone governance](#)

Testing approach for Azure landing zones

Article • 10/09/2023

ⓘ Note

This article only applies to Microsoft Azure and not to any other Microsoft Cloud offerings such as Microsoft 365 or Microsoft Dynamics 365.

Some organizations might want to test their Azure landing zones platform deployment for Azure Policy definitions and assignments, role-based access control (RBAC) custom roles and assignments, and so on. The tests can be completed via automation by using Azure Resource Manager templates (ARM templates), [AzOps](#), [Terraform](#), [Bicep](#), or manually via the Azure portal. This guidance provides an approach that can be used to test changes and their impact in an Azure landing zones platform deployment.

This article can also be used with the [Platform automation and DevOps critical design area](#) guidance as it relates to the PlatformOps and Central functions teams and tasks.

This guidance is most suited to organizations with robust change management processes governing changes to the production environment management group hierarchy. The *canary* management group hierarchy can be independently used to author and test deployments before you deploy them into the production environment.

ⓘ Note

The term *canary* is used to avoid confusion with development environments or test environments. This name is used for illustration purposes only. You might define any name you deem as appropriate for your canary Azure landing zones environment.

Similarly, the term *production environment* is used throughout this guidance to refer to the management group hierarchy your organization might have in place that contains the Azure subscriptions and resources for your workloads.

Platform definition

ⓘ Important

This guidance is not for development environments or test environments that will be used by application or service owners known as landing zones, workloads, applications, or services. These are placed and handled within the production environment management group hierarchy and associated governance (RBAC and Azure Policy).

This guidance is only for platform level testing and changes in the context of Azure landing zones.

Enterprise-scale helps you design and deploy the required Azure platform components to enable you to construct and operationalize landing zones at scale.

The platform resources in scope for this article and this testing approach are:

Product or service	Resource provider and type
Management groups	Microsoft.Management/managementGroups
Management groups subscription association	Microsoft.Management/managementGroups/subscriptions
Policy definitions	Microsoft.Authorization/policyDefinitions
Policy initiative definitions or policy set definitions	Microsoft.Authorization/policySetDefinitions
Policy assignments	Microsoft.Authorization/policyAssignments
RBAC role definitions	Microsoft.Authorization/roleDefinitions
RBAC role assignments	Microsoft.Authorization/roleAssignments
Subscriptions	Microsoft.Subscription/aliases

Example scenarios and outcomes

An example of this scenario is an organization that wants to test the impact and result of a new Azure Policy to govern resources and settings in all landing zones, as per the [Policy-driven governance design principle](#). They don't want to make this change directly to the production environment as they're concerned about the impact it might have.

Using the canary environment to test this platform change will allow the organization to implement and review the impact and result of the Azure Policy change. This process will ensure it satisfies the organization's requirements before they implement the Azure Policy to their production environment.

A similar scenario might be a change to the Azure RBAC role assignments and Microsoft Entra group memberships. It might require a form of testing before the changes are made in the production environment.

ⓘ Important

This is not a common deployment approach or pattern for most customers. It isn't mandatory for Azure landing zones deployments.

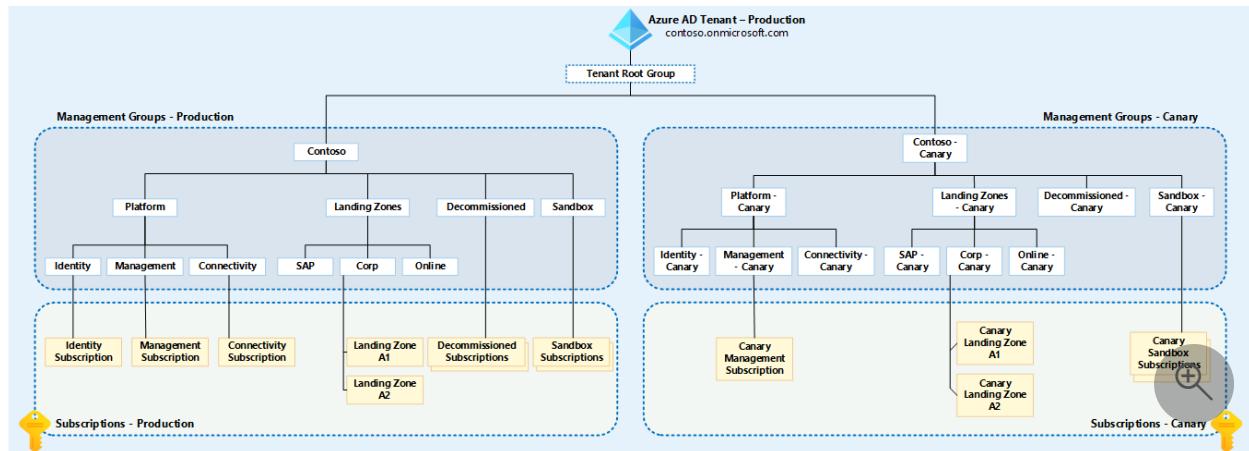


Figure 1: Canary management group hierarchy.

As the diagram shows, the entire Azure landing zones production environment management group hierarchy is duplicated under the `Tenant Root Group`. The *canary* name is appended to the management group display names and IDs. The IDs must be unique within a single Microsoft Entra tenant.

ⓘ Note

The canary environment management group display names can be the same as the production environment management group display names. This might cause confusion for users. Because of this, we recommend to append the name "canary" to the display names, as well as to their IDs.

The canary environment management group hierarchy is then used to simplify testing of the following resource types:

- Management groups
 - Subscription placement
- RBAC
 - Roles (built-in and custom)
 - Assignments

- Azure Policy
 - Definitions (built-in and custom)
 - Initiatives, also known as set definitions
 - Assignments

What if you don't want to deploy the entire canary environment management group hierarchy?

If you don't want to deploy the entire canary environment management group hierarchy, you can test platform resources within the production environment hierarchy by using [sandbox subscriptions](#) as shown in the diagram.

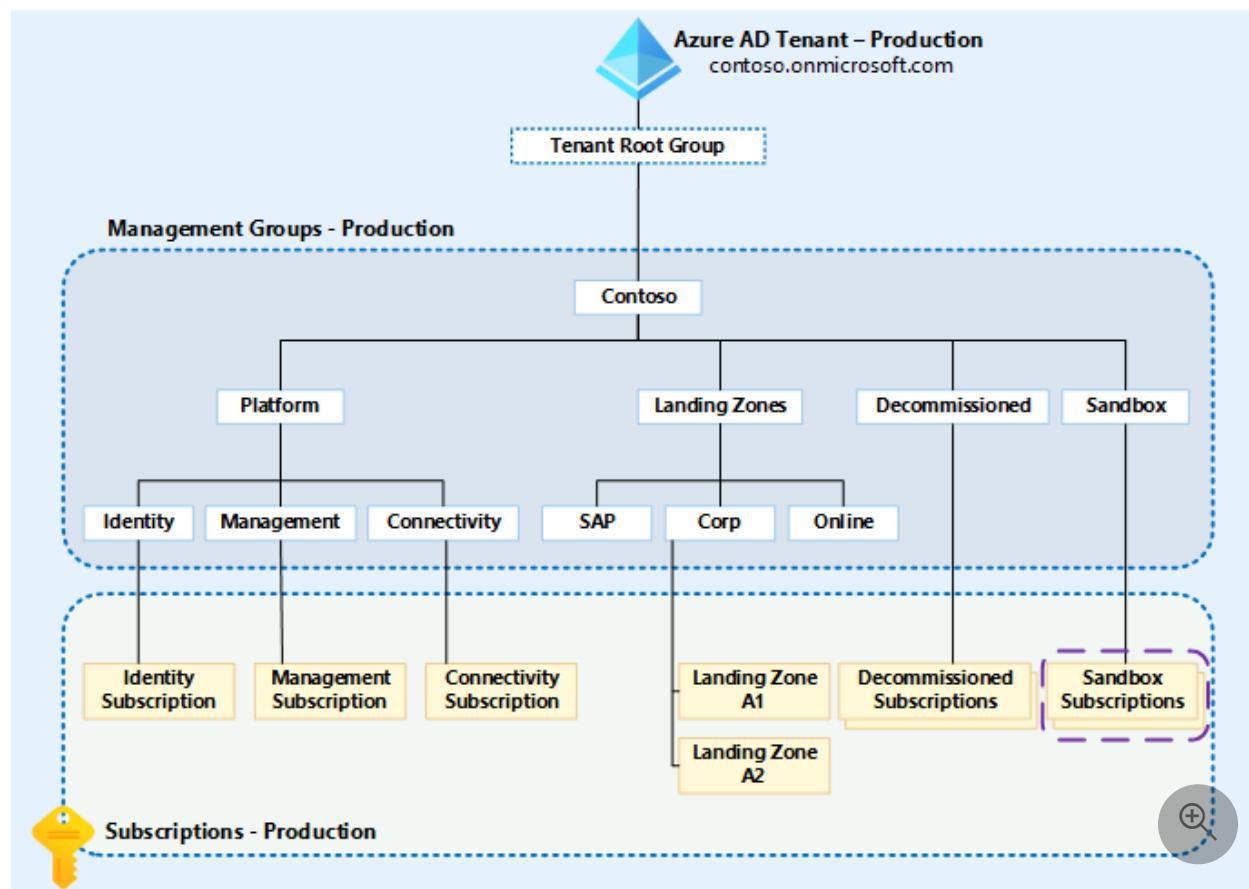


Figure 2: Enterprise-scale management group hierarchy highlighting sandboxes.

To test Azure Policy and RBAC in this scenario, you need a single Azure subscription with the Owner RBAC role assigned to the identity you wish to complete the testing as, for example, User Account, Service Principal, or Managed Service Identity. This configuration will allow you to author, assign, and remediate Azure Policy definitions and assignments within the scope of the sandbox subscription only.

This sandbox approach can also be used for RBAC testing within the subscription, for example, if you're developing a new custom RBAC role to grant permissions for a particular use case. This testing can all be done in the sandbox subscription and tested before you create and assign roles higher up in the hierarchy.

A benefit of this approach is that the sandbox subscriptions can be used for the time that they're required, and then deleted from the environment.

However, this approach doesn't allow you to test with the inheritance of RBAC and Azure policies from the management group hierarchy.

Using a single Microsoft Entra tenant

Considerations to take into account when you use a single Microsoft Entra tenant are:

- Follows [Enterprise-scale design recommendations](#) for Microsoft Entra tenants.
- As per the [Cloud Adoption Framework Azure best practices, standardize on a single directory and identity](#) guidance, single Microsoft Entra tenants are best practice for most.
 - In a single Microsoft Entra tenant, you can use the different Microsoft Entra groups for both production environments and canary Azure landing zones environments, with the same users, assigned to their relevant management group hierarchy within the same Microsoft Entra tenant.
- Increased or duplicated Microsoft Entra ID licensing costs because of multiple identities across different Microsoft Entra tenants.
 - This point is especially relevant to customers who use Microsoft Entra ID P1 or P2 features.
- RBAC changes will be more complex in both canary environments and production environments, as it's likely that the users and groups aren't identical across both Microsoft Entra tenants.
 - Furthermore, the users and groups IDs won't be the same across Microsoft Entra tenants because of them being globally unique.
- Reduces complexity and management overhead caused by managing multiple Microsoft Entra tenants.
 - Privileged users that must maintain access and sign in to separate tenants to perform testing might make changes to the production environment accidentally, instead of making changes to the canary environment and vice versa.
- Reduces the likelihood of configuration drift and deployment failures.
- Doesn't require extra security and break-glass or emergency access processes to be created.

- Reduces friction and the time required to implement changes to the Azure landing zones deployment.

Implementation guidance

Below is guidance on how to implement and use the canary management group hierarchy for Azure landing zones alongside a production environment management group hierarchy.

Warning

If you are using the portal to deploy and manage your Azure landing zones environment today, it might be difficult to adopt and use the canary approach efficiently due to a high risk of both the production and canary environments getting out-of-sync often and therefore not providing a replica-like hierarchy and environment of production.

Consider moving to an Infrastructure-as-Code deployment approach for Azure landing zones, as listed above, if you are in this scenario. Or be aware of the potential risks of configuration drift between canary and production and proceed with care.

1. Use separate Microsoft Entra service principals (SPNs) or Managed Service Identities (MSIs) that are granted permissions over the relevant production environment or canary environment management group hierarchy.
 - This guidance follows the principle of least privilege (PoLP)
2. Use separate folders within a git repository, branches, or repositories to hold the Infrastructure-as-Code for the production environment and canary environment Azure landing zones deployments.
 - Using the relevant Microsoft Entra service principals (SPNs) or Managed Service Identities (MSIs) as part of the CI/CD pipelines depending on which hierarchy is being deployed.
3. Implement git branch policies or security for the canary environment as you have in place for the production environment.
 - Consider reducing the number of approvers and checks for the canary environment to fail-fast.

4. Use the same Azure Pipelines or GitHub actions that use environment variables to change which hierarchy is being deployed. Another option is to clone the pipelines and amend the hard-coded settings to define which hierarchy is being deployed.
 - Using [Azure Pipelines DevOps templates](#) or [GitHub Actions Workflow Templates](#) will help you adhere to the *don't repeat yourself (DRY)* principle.
5. Have a set of canary subscriptions under a separate EA department and account that can be moved around the canary management group hierarchy as needed.
 - It might be beneficial to have a set of resources always deployed into the canary environment subscriptions.
 - It might be helpful to have Infrastructure-as-Code templates such as ARM templates, Bicep, or Terraform, that create a set of resources that enable validation of changes in the canary environment.
6. Send all Azure activity logs for all Azure subscriptions, including any canary environment subscriptions, to the production environment Azure Log Analytics workspace as per the [Azure landing zones design recommendations](#).

💡 Tip

If you already have Azure landing zones deployed in production and are now looking to add a canary environment. Consider cloning your current deployment of the production environment hierarchy and amend the names of resources to prefix them with your canary naming scheme.

This is to ensure what you are deploying to enable the canary environment is in sync with production from the start. This is easily achieved when using an Infrastructure-as-Code tool alongside a git repository.

Next steps

Learn how to implement landing zone sandbox environments.

[Implement landing zone sandbox environments](#)

Landing zone sandbox environments

Article • 06/06/2023

A sandbox is an isolated environment where you can test and experiment without affecting other environments, like production, development, or user acceptance testing (UAT) environments. Conduct proof of concepts (POCs) with Azure resources in a controlled environment. Each sandbox has its own Azure subscription, and Azure policies control the subscription. The policies are applied at the sandbox management group level, and the management group inherits policies from the hierarchy above it. Depending on its purpose, an individual or a team can use a sandbox.

💡 Tip

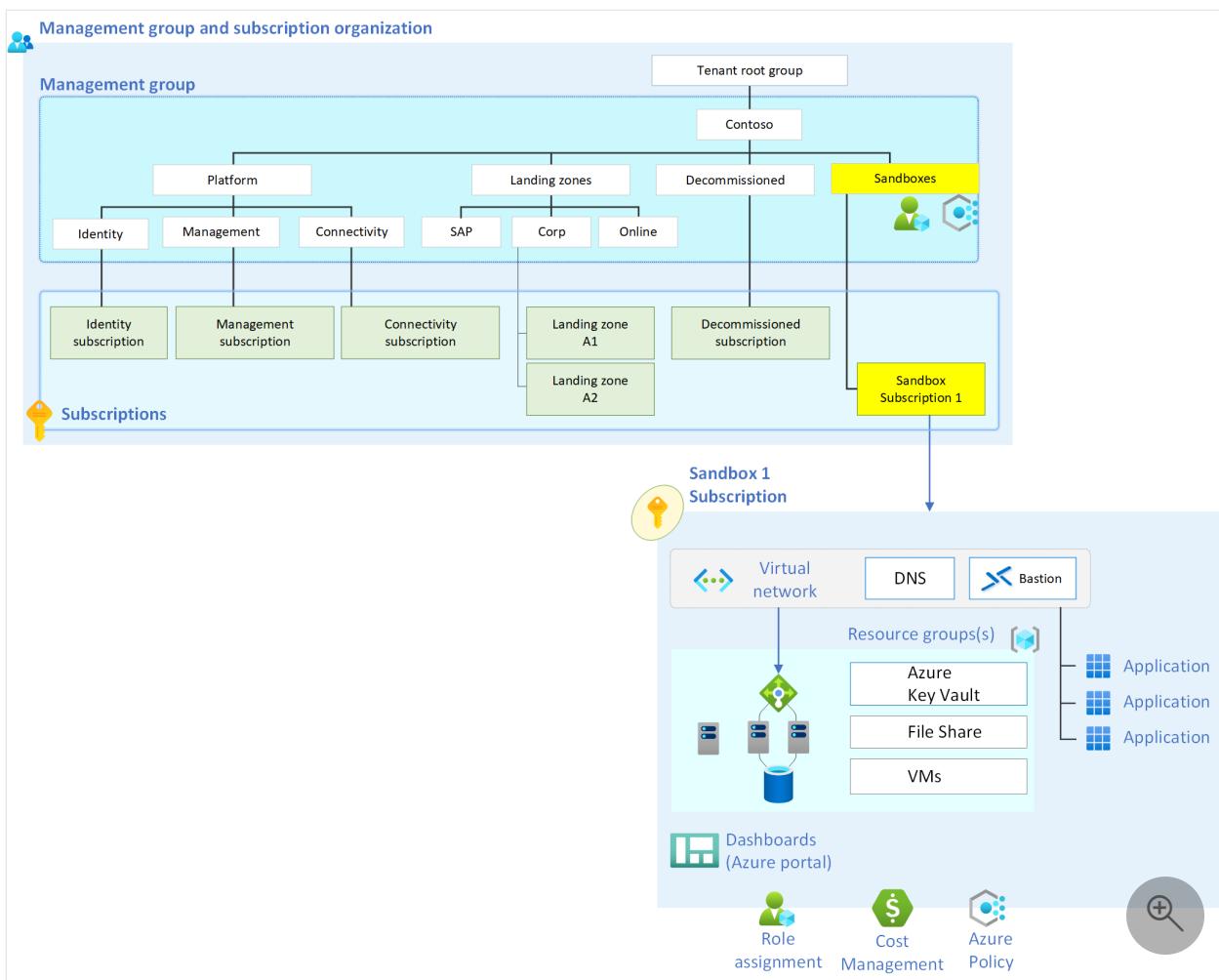
For information about the default Azure landing zones policy assignments, see [Policies included in Azure landing zones reference implementations](#).

Sandbox environments are the best place for hands-on Azure learning. Some common use cases include:

- A developer needs a controlled Azure environment to quickly test application design patterns.
- A cloud architect needs a sandbox environment to evaluate Azure resources or conduct POCs for an Azure service or resource before formally approving them for their organization.
- A cloud engineer needs a sandbox environment to better understand what happens when a setting is changed on an Azure resource.
- A platform engineer wants to build and test out a new Azure policy and see how it behaves per the [Canary guidance](#).
- A developer wants to experiment with Azure services or resources while building an application.

Sandbox architecture

The following image shows the management group and subscription layout.



Place the sandbox subscription in the sandbox management group. For more information about management groups and subscription organization, see [Landing zone design areas and conceptual architecture](#). Azure policies that are created for sandboxes are placed at the management group level of the sandbox. Sandbox environments then inherit Azure policies from the management group hierarchy that's above them.

A sandbox subscription helps manage costs for each program or project. You can easily track costs and cancel sandboxes when budgets decrease or when the sandbox expires.

Networking

Create the sandbox subscription networking that fits your needs. To keep the sandbox isolated, ensure that the networks created inside the sandbox subscriptions haven't peered with other networks outside of the sandbox. You can use the [deny virtual network peering cross subscription](#) policy to ensure that each sandbox is its own isolated environment.

Use the [deny ExpressRoute/VPN/Virtual WAN creation](#) policy to deny the creation of ExpressRoute gateways, VPN gateways, and Virtual WAN hubs. When you deny these resources, it ensures that the sandbox subscription networks remain isolated.

Audit logging

For security, it's important to enable audit logging for a sandbox environment. Enable a diagnostic setting that includes at least the administrative and security log categories (audit) for all sandbox subscriptions. Store audit logs in a central destination like the Azure landing zone default Log Analytics workspace so you can review them easily. Or you can integrate them with a security information and event management (SIEM) platform, like [Microsoft Sentinel](#). For more information, see [Inventory and visibility recommendations](#).

The [Azure policies included in the enterprise-scale landing zone reference implementation](#) have an Azure policy definition ("Configure Azure activity logs to stream to specified Log Analytics workspace") that enables audit logging for all subscriptions. The sandbox management group should inherit this policy to enable sandbox subscription diagnostic logging.

Sandbox access

The sandbox user has owner access to the sandbox subscription. When a sandbox is canceled, remove the owner role-based access control (RBAC) for all sandbox users.

Other considerations

To ensure reliable and efficient sandbox environment performance, consider the following factors.

Sandbox expiration

You can cancel or delete a sandbox when necessary. Plan a strategy for removing sandboxes to save on costs and ensure that security remains dependable. Consider the cost and sandbox expiration date to determine when to remove a sandbox. After a sandbox expires, move it to the [decommissioned](#) management group.

Cost

A key concern for cloud-based sandbox environments is cost tracking. To make tracking easier, you can create a budget in Microsoft Cost Management. The [budgets feature](#) sends you alerts when actual spending or forecasted spending crosses a configured threshold.

When you deploy a sandbox, you can create a Microsoft Cost Management budget and assign it to the subscription. The budget feature alerts the sandbox users when spending thresholds cross the percentage that you specify. For example, you can set an alert for when the budget crosses the 100% spend threshold. In that case, you might want to [cancel](#) or delete a subscription. The alert alone is just a warning mechanism.

You can assign a budget to all sandboxes. Apply a default budget by using the [Deploy-Budget](#) Azure policy at the sandbox management group level. Set the default budget to the maximum cost the organization approves for a sandbox. The default budget sends cost alerts for any sandbox that isn't assigned a more specific budget.

Expiration date

Most organizations want to expire and delete sandboxes after a period of time. Expire sandboxes to provide cost control and security benefits. Sandbox environments are created for testing and learning purposes. After the sandbox user performs their test or gains the intended knowledge, you can expire the sandbox because it's no longer needed. Give an expiration date to each sandbox. When that date is reached, [cancel](#) or delete the sandbox subscription.

When you create a sandbox, you can place an Azure [tag](#) with an expiration date on the subscription. Use automation to cancel or delete the subscription when it reaches the expiration date.

Restrict Azure resources

To provide the most robust learning environment for sandbox users, make all Azure services available in the sandbox environment. Unrestricted sandboxes are ideal, but some organizations have requirements to restrict which Azure services are deployed to sandboxes. Control these restrictions via Azure Policy. Use the [Azure service blocklist](#) policy to deny the deployment of specific Azure services.

Information protection

Most organizations agree that it's important to keep sensitive data out of a sandbox environment. The first line of defense for information protection is user education. Before assigning a user to a sandbox, provide them with disclaimers and information that clearly states not to add sensitive data to the sandbox.

Use [Microsoft Purview](#) to provide information protection for sandbox environments. Purview can send alerts if a user adds data that the organization labels as sensitive to

sandbox environments.

Next steps

Azure sandbox guide

Improve landing zone governance

Landing zone regions

Article • 05/29/2024

This article explains how landing zones use Azure regions. The Azure landing zone architecture is region-agnostic, but you need to specify Azure regions to deploy your Azure landing zone architecture. The following guidance describes how to add a region to an existing landing zone and also provides considerations for when you migrate your Azure estate to a different region.

In some situations, you should deploy applications into multiple Azure regions to support your high availability and disaster recovery business requirements. You might not have an immediate need for multi-region applications, but you should design your Azure landing zone platform to support multiple regions, especially for connectivity, identity, and management services. Ensure that you can quickly enable and support multi-region application landing zones.

For more information, see [Select Azure regions](#).

Landing zones and Azure regions

Azure landing zones consist of a set of resources and configuration. Some of these items, like management groups, policies, and role assignments, are stored at either a tenant or management group level within the Azure landing zone architecture. These resources aren't *deployed* to a particular region and instead are deployed globally. However, you still need to specify a deployment region because Azure tracks some of the resource metadata in a regional metadata store.

Other resources are deployed regionally. Depending on your own landing zone configuration, you might have some or all of the following regionally deployed resources:

- An Azure Monitor Logs workspace, including selected solutions
- An Azure Automation account
- Resource groups, for the other resources

If you deploy a networking topology, you also need to select an Azure region to deploy the networking resources to. This region can be different from the region that you use for the resources listed in the preceding list. Depending on the topology you select, the networking resources that you deploy might include:

- Azure Virtual WAN, including a Virtual WAN hub

- Azure virtual networks
- VPN gateway
- Azure ExpressRoute gateway
- Azure Firewall
- Azure DDoS Protection plans
- Azure private DNS zones, including zones for Azure Private Link
- Resource groups, to contain the preceding resources

 **Note**

To minimize the effect of regional outages, we recommend that you place resources in the same region as the resource group. For more information, see [Resource group location alignment](#).

Add a new region to an existing landing zone

You should consider a multi-region strategy, either from the start of your cloud journey, or by expanding into more Azure regions after you complete the initial deployment of your Azure landing zone architecture. For example, if you use Azure Site Recovery to enable disaster recovery for your virtual machines, you might want to replicate them to a different Azure region. To add Azure regions within your Azure landing zone architecture, consider the following recommendations:

 Expand table

Area	Recommendation
Management groups	No action necessary. Management groups aren't tied to a region. You shouldn't create a management group structure based on regions.
Subscriptions	Subscriptions aren't tied to a region.
Azure Policy	Make changes in Azure Policy if you assigned policies to deny resource deployment to all regions by specifying a list of "allowed" Azure regions. Update these assignments to allow resource deployments to the new region you want to enable.
Role-based access control (RBAC)	No action necessary. Azure RBAC isn't tied to a region.
Monitoring and logging	Decide whether to use a single Azure Monitor Logs workspace for all regions or to create multiple workspaces to cover various geographical regions. Each approach has advantages and disadvantages, including potential cross-region

Area	Recommendation
	networking charges. For more information, see Design a Log Analytics workspace architecture .
Networking	<p>If you deploy a networking topology, Virtual WAN, or traditional hub and spoke, expand the networking to the new Azure region. Create another networking hub by deploying the required networking resources into the existing Connectivity subscription in the new Azure region. Azure Virtual Network Manager can make it easier to expand and manage virtual networks at scale in multiple regions.</p> <p>From a Domain Name System (DNS) perspective, you might also want to deploy forwarders, if you use them, into the new Azure region. Use forwarders for spoke virtual networks in the new region for resolution. Azure DNS zones are global resources and not tied to a single Azure region, so they don't require any action.</p>
Identity	<p>If you deployed Active Directory Domain Services or Microsoft Entra Domain Services into your <i>Identity</i> subscription or spoke, expand the service into the new Azure region.</p>

ⓘ Note

You should also use [availability zones](#) for high availability within a region. Check whether your [Azure regions support availability zones](#), and [how the services you use support availability zones](#).

Microsoft Cloud for Sovereignty has guidelines for restricting services and regions. You can use these guidelines to enforce service configuration to help customers achieve their [data residency](#) needs.

High-level approach

When you expand an Azure landing zone into a new region, consider following the steps in these sections. Before you start, you need to decide on a new Azure region, or multiple Azure regions, to expand into.

Networking

Traditional hub-and-spoke architecture

ⓘ Tip

See a [traditional hub-and-spoke architecture](#).

1. Decide on whether you need a new platform landing zone subscription. We recommend that most customers use their existing *Connectivity* subscriptions, even when they use multiple regions.
2. Within the subscription, create a new resource group in the new target region.
3. Create a new hub virtual network in the new target region.
4. If applicable, deploy Azure Firewall or network virtual appliances (NVAs) into your new hub virtual network.
5. If applicable, deploy virtual network gateways for VPN or ExpressRoute connectivity, and establish connections. Ensure that your ExpressRoute circuits and on-premises locations follow Microsoft resiliency recommendations. For more information, see [Designing for disaster recovery with ExpressRoute private peering](#).
6. Establish virtual network peering between the new hub virtual network and the other hub virtual networks.
7. Create and configure any required routing, such as Azure Route Server or user-defined routes.
8. If required, deploy DNS forwarders for the new target region and link to any private DNS zones to enable name resolution.
 - Some customers might configure name resolution on their Windows Server Active Directory domain controllers within the *Identity* platform landing zone subscription.

To host your workloads, you can then use virtual network peering to connect application landing zone spokes to the new hub virtual network in the new region.

 **Tip**

[Virtual Network Manager](#) can make it easier to expand and manage virtual networks at scale in multiple regions.

Virtual WAN architecture

 **Tip**

See a [Virtual WAN architecture](#).

1. Within your existing Virtual WAN, create a new virtual hub in the new target region.
2. Deploy Azure Firewall or other supported network virtual appliances (NVAs) into your new virtual hub. Configure [Virtual WAN routing intent and routing policies](#) to route traffic through the new secured virtual hub.
3. If applicable, deploy virtual network gateways for VPN or ExpressRoute connectivity in the new virtual hub and establish connections. Ensure that your ExpressRoute circuits and on-premises locations follow Microsoft resiliency recommendations. For more information, see [Designing for disaster recovery with ExpressRoute private peering](#).
4. Create and configure any other routing that you require, such as virtual hub static routes.
5. If applicable, deploy DNS forwarders for the new target region and link to any private DNS zones to enable resolution.
 - Some customers might configure name resolution on their Active Directory domain controllers within the *Identity* platform landing zone subscription.
 - In Virtual WAN deployments, this must be in a spoke virtual network that is connected to the virtual hub through a virtual network connection, following the [Virtual hub extension pattern](#).

To host your workloads, you can then use virtual network peering to connect application landing zone spokes to the new hub virtual network in the new region.

Identity

Tip

Review the Azure landing zone design area for [identity and access management](#).

1. Decide whether you need a new platform landing zone subscription. We recommend that most customers use their existing *Identity* subscription, even when they use multiple regions.
2. Create a new resource group in the new target region.

3. Create a new virtual network in the new target region.
4. Establish virtual network peering back to the newly created regional hub virtual network in the *Connectivity* subscription.
5. Deploy identity workloads, like Active Directory domain controller virtual machines, into the new virtual network.
 - You might need to perform more setup of the workloads once they're provisioned, such as the following configuration steps:
 - Promote the Active Directory domain controller virtual machines to the existing Active Directory domain.
 - Create new Active Directory sites and subnets.
 - Configure DNS server settings like conditional forwarders.

Move your Azure estate to a new region

You might occasionally need to move your entire Azure estate to a different region. For example, suppose you deployed your landing zone and workloads into an Azure region in a neighboring country or region, and then a new Azure region launches in your home country or region. You might elect to move all of your workloads to the new region to improve the communication latency, or to comply with data residency requirements.

Note

This article provides information about migrating the landing zone components of your estate. For more information, see [Relocate cloud workloads](#).

Global landing zone configuration

Most of the globally deployed landing zone configuration doesn't typically need to be modified when you move regions. However, ensure that you check for any policy assignments that restrict region deployments and update the policy to allow deployments into the new region.

Regional landing zone resources

Region-specific landing zone resources often require more consideration because you can't move some Azure resources between regions. Consider the following approach:

1. **Add the destination region as an additional region to your landing zone:** For more information, see [Add a new region to an existing landing zone](#).
2. **Deploy centralized components in the destination region:** For example, deploy a new Azure Monitor Logs workspace in your destination region so that workloads can begin to use the new component after you migrate the workload.
3. **Migrate your workloads from the source region to the destination region:** During the workload migration process, reconfigure the resources to use the destination region's networking components, identity components, the Azure Monitor Logs workspace, and other regional resources.
4. **Decommission the resources in the source region after you complete the migration.**

Consider the following tips when you migrate landing zone resources across regions:

- **Use infrastructure as code:** Use Bicep, Azure Resource Manager templates (ARM templates), Terraform, scripting, or a similar approach to export and reimport complex configurations, such as rules for Azure Firewall.
- **Automation:** Use [scripts](#) to migrate Automation resources between regions.
- **ExpressRoute:** Consider whether you can use the [ExpressRoute Local SKU](#) in your destination region. If your on-premises environments are within the same metropolitan area as your Azure region, ExpressRoute Local can provide a lower-cost option compared to other ExpressRoute SKUs.

Next step

[Improve landing zone governance](#)

Feedback

Was this page helpful?

 Yes

 No

Advanced Azure Policy management

Article • 01/05/2024

This article describes how to manage Azure Policy at scale by using infrastructure as code (IaC). Policy-driven governance is a design principle for Azure landing zones. It helps to ensure that the applications you deploy comply with your organization's platform. It can take considerable effort to manage and test policy objects across an environment to ensure that compliance is met. [Azure landing zone accelerators](#) help to establish a secure baseline, but your organization might have further compliance requirements that you must meet by deploying other policies.

What is Enterprise Policy as Code (EPAC)?

[EPAC](#) is an open-source project that you can use to integrate IaC and manage Azure Policy. EPAC is built upon a PowerShell module and published to the PowerShell Gallery. You can use the features of this project to:

- Create stateful policy deployments. The objects that are defined in the code become the source of truth for policy objects deployed in Azure.
- Implement complex policy management scenarios, such as multitenant and sovereign-cloud deployments.
- Export and integrate policies to incorporate existing custom policies that were developed prior to the Azure landing zone deployment.
- Create and manage policy exemptions and policy documentation.
- Use sample workflows to demonstrate Azure Policy deployments with GitHub Actions or Azure Pipelines.
- Export noncompliance reports and create remediation tasks.

Reasons to use EPAC

You can use EPAC to deploy and manage Azure landing zone policies. You might want to consider implementing EPAC to manage policies if:

- You have unmanaged policies in an existing brownfield environment that you want to deploy in a new Azure landing zone environment. [Export the existing policies](#), and manage them with EPAC alongside the Azure landing zone policy objects.

- You have an Azure deployment that doesn't fully align to an Azure landing zone, for example multiple management group structures for testing or a nonconventional management group structure. The default assignment structure that other Azure landing zone deployment methods provide might not fit your strategy.
- You have a team that's not responsible for infrastructure deployment, for example a security team that might want to deploy and manage policies.
- You require features from policies that aren't available in the Azure landing zone accelerator deployments, for example policy exemptions and documentation.

Get started

The EPAC GitHub repository provides detailed steps to start managing Azure Policy. Consider the following factors when determining whether the project is a good fit for your environment:

- *Environment topology*: Multiple tenancies and complicated management group structures are supported. Consider how you want to structure your policy as code deployments to fit the topology, so multiple teams can manage policies and test new policy deployments.
- *Permissions*: Consider how you manage permissions for the deployment, especially for roles and identities. EPAC provides multiple stages to deploy both the policies and role assignments, so separate identities can be used.
- *Existing policy deployments*: In a brownfield scenario, you might have existing policies that must remain in place while EPAC is deployed. You can use the [desired state strategy](#) to ensure that EPAC manages only the defined policies and preserves existing policies.
- *Deployment methodology*: EPAC supports Azure DevOps, GitHub Actions, and a PowerShell module to help deploy policies. You can use the [sample pipelines in the EPAC starter kit](#) and adapt them to your environment and requirements.

Follow the [quickstart guide](#) to export policy objects in your environment and get familiar with how EPAC manages Azure Policy.

For issues with the code or documentation, [submit an issue in the GitHub repository](#).

Replace existing policy deployment solutions

EPAC replaces the policy deployment capabilities of the Azure landing zone accelerators. When you use these accelerators, you shouldn't use them to deploy Azure Policy because EPAC is the source of truth for policy in the environment.

For more information, see the following resources for policy management with Bicep and Terraform Azure landing zone accelerators:

- [How Does ALZ-Bicep implement Azure policies?](#)
- [Archetype definitions](#)

Next steps

- [Enterprise Policy as Code](#)
-

Feedback

Was this page helpful?

 Yes

 No

Improve landing zone governance

Article • 04/05/2024

As your cloud environment evolves over time, you need to evolve your cloud governance policies and processes that apply to your landing zones. For more information on cloud governance, see the [Govern methodology](#).

Landing zone governance best practices

The following best practices provide examples of ways to improve landing zone governance:

- [Naming and tagging standards](#): Ensure consistency in naming and tagging, which is the foundational data for establishing sound governance practices.
- [Track costs across workloads](#): Begin tracking costs in your first landing zone. Evaluate how you track costs across multiple workloads and roles.
- [Scale with multiple subscriptions](#): Evaluate how this landing zone and other landing zones scale as you add more subscriptions.
- [Organize subscriptions](#): Understand how to organize and manage multiple subscriptions.
- [Keep your Azure landing zone up to date](#): Why you should update your Azure landing zone environment, and how to get more information.

Next steps

The cloud environment expands with each new workload you add. To stay ahead of these requirements, cloud platform teams should periodically review the landing zone design areas.

[Review landing zone design areas](#)

Feedback

Was this page helpful?

 Yes

 No

Develop your naming and tagging strategy for Azure resources

Article • 03/22/2023

Organize your cloud assets to support governance, operational management, and accounting requirements. Well-defined naming and metadata tagging conventions help to quickly locate and manage resources. These conventions also help associate cloud usage costs with business teams via chargeback and showback accounting mechanisms.

Define your naming and tagging strategy as early as possible. Use the following links to help you define and implement your strategy:

- [Define your naming convention](#)
- [Recommended abbreviations for Azure resource types](#)
- [Define your tagging strategy](#)
- [Resource naming and tagging decision guide](#)
- [Naming rules and restrictions for Azure resources](#)

ⓘ Note

Every business has its own organizational and management requirements. These recommendations help start a discussion with your cloud adoption teams. As the discussion proceeds, use the tools below to document the naming and tagging decisions you make when aligning these recommendations to your specific business needs.

Download the [Azure Naming Tool](#) to create an organizational naming reference and name generator.

Download the [naming and tagging conventions tracking template](#).

Purpose of naming and tagging

Accurately representing and naming your resources is essential for security purposes. If you come upon a security incident, it's critical to quickly identify affected systems, what functions those systems support, and the potential business impact. Security services such as [Microsoft Defender for Cloud](#) and [Microsoft Sentinel](#) reference resources and their associated logging and alert information by resource name.

Azure defines [naming rules and restrictions for Azure resources](#). This guidance provides you with detailed recommendations to support enterprise cloud adoption efforts.

Changing resource names can be difficult. Establish a comprehensive naming convention before you begin any large cloud deployment.

Naming and tagging strategy

A naming and tagging strategy includes business and operational details as components of resource names and metadata tags:

- The business side of this strategy ensures that resource names and tags include the organizational information you need to identify the teams. Use a resource along with the business owners who are responsible for resource costs.
- The operational side ensures that names and tags include necessary information. IT teams use this information to identify the workload, application, environment, criticality, and other information useful for managing resources.

Next steps

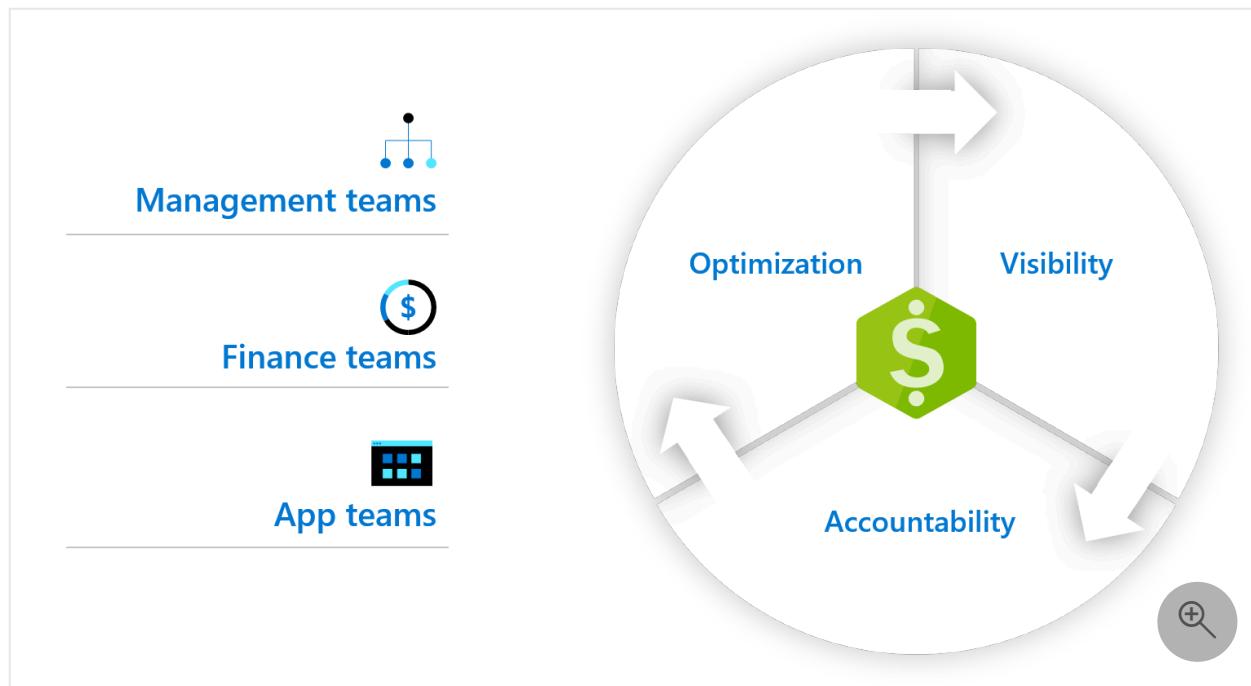
Learn about the considerations for defining your naming convention of your Azure resources and assets, and review example names for resources and assets in Azure.

[Name your Azure resources and assets](#)

Track costs across business units, environments, or projects

Article • 05/14/2024

To build a [cost-conscious organization](#), you need visibility and properly defined access to cost-related data. This best-practice article outlines decisions and implementation approaches to help you create tracking mechanisms to monitor costs. You'll learn how to apply fundamental Azure concepts to provide cost visibility.



Establish a well-managed environment

Cost control, much like governance and other management constructs, depends on a well-managed environment. To establish such an environment, especially a complex one, you need to consistently classify and organize all assets. Azure provides several mechanisms for classifying and organizing assets.

Assets, which are also known as resources, include all virtual machines, data sources, and applications deployed to the cloud. [Organize and manage your subscriptions](#) based on multiple criteria to establish a well-managed environment.

Classify assets

Tagging is an easy way to classify assets. Tagging associates metadata to an asset. That metadata can be used to classify the asset based on various data points. Tagging is a

fundamental part of any well-managed environment, and it's necessary for establishing proper governance of any environment.

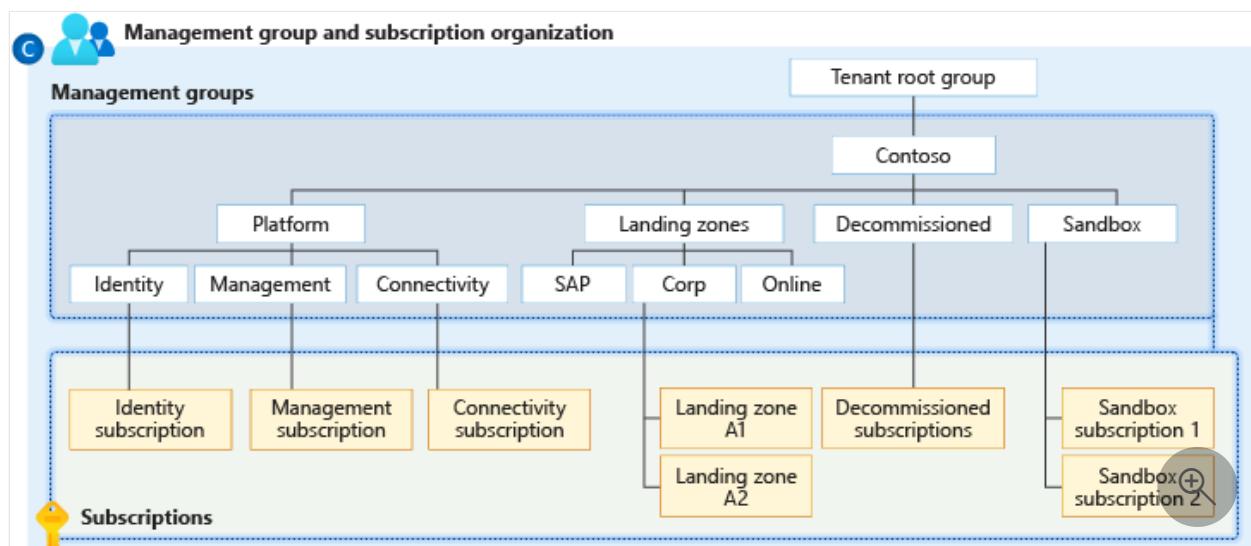
The first step is to [develop naming and tagging standards](#). The second step is to ensure that the tagging standard is consistently applied by [establishing a governance MVP](#).

When tags are used to classify assets as part of a cost management effort, companies often need the following tags: business unit, department, billing code, geography, environment, project, and workload or application categorization. [Microsoft Cost Management](#), a tool for setting budgets and gaining visibility into cloud costs for Azure or AWS, can use these tags to create different views of cost data.

Organize assets

There are several approaches to organizing assets. Microsoft's enterprise-scale [Azure landing zone](#) design provides an architecture that can be used as the basis of any Azure cloud environment. The landing zone [resource organization](#) documentation provides detailed guidance on organizing [management groups](#) and [subscriptions](#). Understanding the [design principles](#) used in designing the conceptual architecture will give you a foundation in best practices as you adapt the architecture to meet your specific business needs. Deviations to the design may be necessary to meet your business requirements, but understanding the impact of those deviations will prepare you for any necessary mitigations.

The following model for management groups, subscriptions, and resource groups creates a hierarchy that provides each team with the right level of visibility to perform their duties. When the enterprise needs cost controls to prevent budget overrun, it can apply governance tooling like Azure Policy to the subscriptions within this structure to quickly block future cost errors.



The rest of this article assumes the use of the best-practice approach in the preceding diagram. But the following articles can help you apply the approach to a resource organization that best fits your company:

- [Scale your Azure environment with multiple subscriptions](#)
- [Organize and manage your Azure subscriptions](#)
- [Deploy a governance MVP to govern well-managed environment standards](#)

Provide the right level of cost access

Managing cost is a team activity. The organization readiness section of the Cloud Adoption Framework defines a few core teams and outlines how those teams support cloud adoption efforts.

For the proper level of visibility into cost management data, the members of the team are assigned scope and roles. **Roles** define what a user can do to various assets. The **Scope** defines which assets, such as user, group, service principal, or managed identity, that a user can affect. As a general best practice, we suggest a least-privilege model in assigning people to various roles and scopes.

Roles

Cost Management supports the following built-in roles for each scope:

- **Owner:** Can view costs and manage everything, including cost configuration.
- **Contributor:** Can view costs and manage everything, including cost configuration, but excluding access control.
- **Reader:** Can view everything, including cost data and configuration, but can't make changes.
- [**Cost Management Contributor:**](#) Can view costs and manage cost configuration.
- [**Cost Management Reader:**](#) Can view cost data and configuration.

As a general best practice, members of all teams should be assigned the role of Cost Management Contributor. This role grants access to create and manage budgets to more effectively monitor and report on costs. But members of the [cloud strategy team](#) should be set to Cost Management Reader only, because they're not involved in setting budgets within the Cost Management tool.

Scope

The following scope and role settings will create the required visibility into cost management. This best practice might require minor changes to align to asset

organization decisions.

- **Cloud adoption team.** As cloud adoption teams primarily focus on implementation of cloud technologies, cost management access to production environments is not typically required. By virtue of normally having Contributor access to non-production or Sandbox subscriptions, this team would inherently have access to cost management data for those subscriptions.
- **Cloud strategy team.** Responsibilities for tracking costs across multiple projects and business units require [Cost Management Reader](#) access at the root level of the management group hierarchy.
 - Assign Cost Management Reader access to this team at the management group, which ensures ongoing visibility into all deployments associated with the subscriptions governed by that management group hierarchy.
- **Cloud governance team.** Responsibilities for managing cost, budget alignment, and reporting across all adoption efforts requires [Cost Management Contributor](#) access at the root level of the management group hierarchy.
 - In a well-managed environment, the cloud governance team likely has a higher degree of access already, making additional scope assignment for Cost Management Contributor unnecessary.
- **Cloud center of excellence.** Responsibility for managing costs related to shared services requires [Cost Management Contributor](#) access at the subscription level. Additionally, this team might require Cost Management Contributor access to resource groups or subscriptions that contain assets deployed by CCoE automation processes to understand how those processes affect costs.
 - **Shared services.** When a cloud center of excellence is engaged, best practice suggests that assets managed by the CCoE are supported from a centralized shared service subscription within a hub and spoke model. In this scenario, the CCoE likely has Contributor or Owner access to that subscription, making additional scope assignment for Cost Management Contributor unnecessary.
 - **CCoE automation/controls.** The CCoE commonly provides controls and automated deployment scripts to cloud adoption teams. The CCoE has a responsibility to understand how these accelerators affect costs. To gain that visibility, the team needs Cost Management Contributor access to any resource groups or subscriptions running those accelerators.
- **Cloud operations team.** Responsibility for managing ongoing costs of production environments requires [Cost Management Contributor](#) access to the Landing Zone and Platform management group nodes.

- The general recommendation puts production and nonproduction assets in separate subscriptions that are governed by nodes of the management group hierarchy associated with production environments. In a well-managed environment, members of the operations team likely have Owner or Contributor access to production subscriptions already, making the Cost Management Contributor role unnecessary.

Extra cost management resources

After you establish access to a well-managed environment hierarchy, the following articles can help you use that tool to monitor and control costs.

Use Cost Management

- Create and manage budgets
- Export cost data
- Optimize costs based on recommendations
- Use cost alerts to monitor usage and spending

Use Cost Management to govern AWS costs

- Set up AWS Cost and Usage Reports integration
- Manage AWS costs

Establish access, roles, and scope

- Understanding cost management scope
- Setting scope for a resource group

Next steps

To get started with Cost Management, see [How to optimize your cloud investment with Cost Management](#).

Feedback

Was this page helpful?

 Yes

 No

Create additional subscriptions to scale your Azure environment

Article • 01/10/2023

Organizations often use multiple Azure subscriptions to avoid per-subscription resource limits and to better manage and govern their Azure resources. It's important to define a strategy for scaling your subscriptions.

ⓘ Note

We recommend that organizations consider the Azure landing zone guidance for **resource organization** as a first step to planning subscriptions within an Azure environment to ensure the broader context of an environment intended to scale is considered

Review fundamental concepts

As you expand your Azure environment beyond your [initial subscriptions](#), it's important to understand Azure concepts such as accounts, tenants, directories, and subscriptions. For more information, see [Azure fundamental concepts](#).

Other considerations might necessitate additional subscriptions. Keep the following in mind as you expand your cloud estate.

Technical considerations

Subscription limits: Subscriptions have defined limits for some resource types. For example, the number of virtual networks in a subscription is limited. When a subscription approaches these limits, you'll need to create another subscription and put additional resources there. For more information, see [Azure subscription and service limits](#).

Classic model resources: If you've been using Azure for a long time, you may have resources that were created using the classic deployment model. Azure policies, Azure role-based access control, resource grouping, and tags cannot be applied to classic model resources. You should move these resources into subscriptions that contain only classic model resources.

Costs: There might be some additional costs for data ingress and egress between subscriptions.

Business priorities

Your business priorities might lead you to create additional subscriptions. These priorities include:

- Innovation
- Migration
- Cost
- Operations
- Security
- Governance

For other considerations about scaling your subscriptions, review the [subscription organization and governance recommendations](#) in the Cloud Adoption Framework.

Moving resources between subscriptions

As your subscription model grows, you might decide that some resources belong in other subscriptions. Many types of resources can be moved between subscriptions. You can also use automated deployments to re-create resources in another subscription. For more information, see [Move Azure resources to another resource group or subscription](#).

Tips for creating new subscriptions

- Identify who is responsible for creating new subscriptions.
- Decide which resource types are available in a subscription by default.
- Decide what all standard subscriptions should look like. Considerations include Azure RBAC access, policies, tags, and infrastructure resources.
- If possible, [programmatically create new subscriptions](#) via a service principal. You must [grant permission to the service principal](#) to create subscriptions. Define a security group that can request new subscriptions via an automated workflow.
- If you're an Enterprise Agreement (EA) customer, ask Azure Support to block creation of non-EA subscriptions for your organization.

Next steps

Create a management group hierarchy to help [organize and manage your subscriptions and resources](#).

[Organize and manage your subscriptions and resources](#)

Resource organization

Article • 05/29/2024

Use the resource organization design area to establish consistent patterns when you organize resources that you deploy to the cloud.

Design area review

Involved roles or functions: This design area requires support from one or more [cloud platform](#) and [cloud center of excellence](#) functions to make and implement decisions.

Scope: Resource organization decisions provide a foundation for all compliance-related design areas. When you plan your resource organization, you can establish consistent patterns for the following areas:

- Naming
- Tagging
- Subscription design
- Management group design

The initial scope of this exercise assumes a subscription design that aligns with the Azure landing zone conceptual architecture. Workload-level or application-level subscription and landing zone assignment supports separation of duties and subscription democratization requirements.

The following assumptions are the basis for workload subscription design pattern guidance:

- Your enterprise commits to long-term cloud operations.
- You need cloud management, security, and governance tooling to manage Azure, hybrid, or multicloud solutions.
- You have management or platform deployments in subscriptions and management groups that are separate from workload or application resources.

Multiple regions: The performance, reliability, and compliance of your cloud-based applications rely on Azure regions. Use the Azure global infrastructure to scale your applications when you need to. Regions provide the capacity to handle varying workloads. Whether you launch a new product or expand your user base, you must have the right resources in the right region to ensure agility, scalability, and high resiliency.

Use multiple regions for critical applications and services that require geo-disaster recovery capabilities. Multiple regions provide maximum resiliency. For information about how to select and operate in multiple regions, see [Select Azure regions](#).

Also consider the following factors when you deploy your workload in multiple regions:

- You can initially deploy in a single region and then expand to [multiple regions](#) in the future.
- To ensure consistency and manageability, properly organize resources when you adopt a multiregion design.
- Depending on your requirements and desired governance model, you can organize multiregion resources at various levels, such as the [management group](#), [subscription and resource group](#), [naming convention](#), and [tagging](#) levels.

New cloud environment: Start your cloud journey with a small set of subscriptions. For more information, see [Create your initial Azure subscriptions](#).

Existing cloud environment: If you have an existing cloud environment, consider the following guidance:

- If your current environment doesn't use [management groups](#), consider incorporating them. You can use management groups to manage policies, access, and compliance across subscriptions at scale.
- If your current environment uses management groups, see [Management groups](#). Use this guidance to help evaluate your implementation.
- If you have existing subscriptions in your current environment, ensure that you use them effectively. Subscriptions act as policy and management boundaries and scale units. For more information, see [Subscriptions](#).
- If you have existing resources in your current environment, see [Naming and tagging](#). Use this guidance to influence your tagging strategy and your naming conventions going forward.
- Use [Azure Policy](#) to establish and enforce consistency with taxonomic tags.

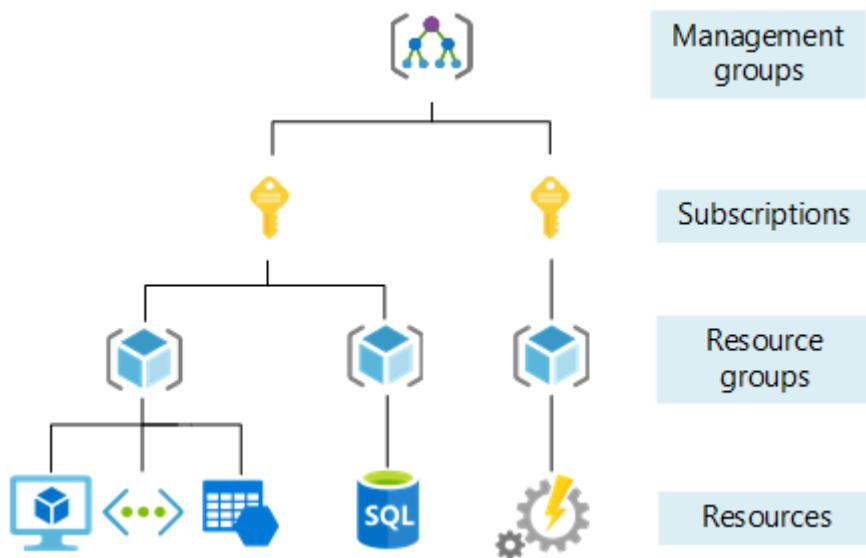
Design area overview

Cloud adoption journeys have various starting points and scale requirements. Some enterprises start with a few applications in the cloud and grow over time. Other enterprises must scale quickly to address business demands like a datacenter migration.

In both scenarios, when you plan your resource organization, you must factor in environment growth to accommodate more applications and services.

Simplify resource management across your environment to prepare for increased workload numbers and scale. Consider the foundational management groups and the subscription structure in your Azure landing zone design and implementation. Plan in advance to prevent future scaling constraints.

The resource organization design area explores techniques and technologies that help maintain proper resource topologies in cloud environments. The following diagram shows the four scope levels to organize Azure resources: management groups, subscriptions, resource groups, and resources.



Next steps

- [Management groups](#)
- [Subscriptions](#)

Feedback

Was this page helpful?

Yes

No

Improve landing zone security

Article • 07/03/2023

When a workload or the landing zones that hosts it require access to any sensitive data or critical systems, it's important to protect the data and assets.

Secure

As you're exiting the Ready state, you have the ongoing responsibility to maintain the security of your environment. Cloud security is also an incremental process instead of just a static destination. Focus on objectives and key results when envisioning a security end state. Map concepts, frameworks, and standards to the disciplines in the CAF [secure methodology](#) alongside mapping to roles and responsibilities for human discipline. The secure methodology provides guidance.

Below we provide an overview of this guidance with links to the details.

Risk insights

Business operations have security [risks](#). The security team should inform and advise decision makers on how security risks fit into their frameworks by understanding the business and using security practices to recognize which risk to appropriately plan and take action upon.

- [What is cybersecurity risk?](#): All potential damage or destruction of the business caused by human attackers attempting to steal currency, inside information, or technology.
- [Align your security risk management](#): Invest in bridging cybersecurity and organizational leadership to explain security threats using business-friendly terminology, actively listening and communicating to all people across the business.
- [Understanding cybersecurity risk](#): Comprehend the motivations and behavior patterns of human attackers for stealing money, information, or technology and identifying potential impact of different types of attacks.

Security integration

Ensure that security is an organizational concern and not siloed into a single group. [Security integration](#) provides you with guidance on how to integrate security into

everyone's role while minimizing friction with business processes. Specific guidance includes:

- **Normalizing relations:** Ensure all teams are integrated with security teams and have a shared understanding of security goals. Further, work to find the right level of security controls, ensuring the controls don't outweigh business value.
- **Integrate with IT and business operations:** Balance the implementation of security updates and mapping how all security processes affect current business impact, and potential security risk in the future.
- **Integrate security teams:** Avoid operating in silos by responding to active threats and continuously improving the security posture of the organization by practicing security as a dynamic discipline.

Business resilience

While organizations can never have perfect security, there's still the pragmatic approach of **Business resilience** in investing the full lifecycle of a security risk before, during, and after an incident.

- **Resilience goals:** Focus on enabling your business to rapidly innovate, limit the impact, and always seek safe ways to adopt technology.
- **Security resilience and assume breach:** Assume breach or compromise to follow the key principle of zero trust and practice pragmatic security behaviors to prevent attacks, limit damage, and have quick recovery from them.

Access control

Make an **access control** strategy that aligns both user experience and security assurances.

- **From security perimeter to zero trust:** Embrace a zero trust approach for access control for establishing and improving security assurances when working in the cloud and using new technology.
- **Modern access control:** Make an access control strategy that is comprehensive, consistent, and flexible. Go beyond a single tactic or technology for multiple workloads, clouds, and various business sensitivity levels.
- **Known, trusted, allowed:** Follow the dynamic three-step process to ensure known authentication, trusting the user or device, and allowing the appropriate rights and privileges for the application, service, or data.
- **Data-driven access decisions:** Make informed decisions from the diverse data on the users and devices for fulfilling explicit validation.

- **Segmentation: Separate to protect:** Create boundaries as separate segments of an internal environment to contain damages of successful attacks.
- **Isolation: Avoid firewall and forget:** Design an extreme form of segmentation for business-critical assets that consists of: people, process, and technology.

Security operations

Establish **security operations** by reducing risk, rapidly responding, and recovery to protect your organization and follow the security discipline of the DevOps process.

- **People and process:** Create a culture to empower people with tools to enable them as your most valuable asset and diversify your thinking portfolio by including and training non-technical people with strong backgrounds in forensic investigation roles.
- **Security operations model:** Focus on the outcomes of incident management, incident preparation, and threat intelligence. Delegate the outcomes between subteams to triage, investigate, and hunt on high volume and complex incidents.
- **SecOps business touchpoints:** Interact with business leadership in order to inform major incidents and determine impact of critical systems. Continuously joint practice response to reduce organizational risk.
- **SecOps modernization:** Evolve security operations by following trends involving platform coverage, identity-centric security, IoT and OT devices, and relevant telemetry from the cloud.

Asset protection

Secure business critical **assets**, which include all physical and virtual items by implementing security controls that are unique to each asset type. Consistently execute preventive and detective protection to meet policies, standards and architecture.

- **Get secure:** Bring resources up to your organization's latest security standards and policy by applying current controls to brownfield assets and ensuring greenfield assets are set to the most recent standards.
- **Stay secure:** Practice continuous cloud improvement and plan for upgrading or retiring end-of-life software as business, technology, and security requirements change rapidly.
- **Get started:** Start protecting assets by focusing on well-known cloud resources first and use well-known and proven vendor/industry baselines for your security configuration.
- **Key information:** Use key elements of accountable and responsible teams to manage enterprise-wide assets such as cloud elasticity workload needs and design

controls to identify best practices. Measure business value of asset protection and favor automated policy to avoid cost and manual repetition.

Security governance

Perform oversight and monitoring with [security governance](#) for sustaining and improving security posture over time by using business goals and risk to determine the best direction for security.

- [Compliance and reporting](#): Have both external and internal security policies meet mandatory requirements in a given industry.
- [Architecture and standards](#): Create a unified view across your enterprise estate as most enterprises are a hybrid environment that includes both on-premises and cloud resources.
- [Security posture management](#): Plan for governance to monitor security standards, provide guidance, and improve processes. Maintain agility by driven governance through policy and continuous improvement.
- [Governance and protection disciplines](#): Apply security controls and provide feedback to identify the best solutions.
- [Governance and security operations](#): Ensure that lessons learned from incidents are integrated into security operations and governance.

Innovation security

Protect the processes and data of innovation against cyberattacks as new applications are developed with [innovation security](#) in mind.

- [What is DevSecOps?](#): Integrated security into the already combined process of development and operations in DevOps to mitigate risks in the innovation process.
- [Secure by design and shifting left](#): Involve security in all stages of the DevOps lifecycle and have teams align with innovation speed, reliability and resilience.
- [Why DevSecOps?](#): To secure the DevOps process protecting against attackers exploiting weaknesses in all IT infrastructure within your organization, which in turn protects your customers.
- [The DevSecOps Journey](#): Use idea incubation and DevOps as a two-phase process like most organizations. Identify the MVP (minimum viable product) requirements, use leadership techniques to resolve teams conflicts, and integrate security in existing processes and tools.
- [Tips on navigating the journey](#): As you transform your security, there will be common challenges throughout the journey that will involve education, time, resourcing, and the overall shifting nature of IT operations.

DevSecOps controls

Add security to each stage of continuous integration and continuous delivery (CI/CD) when making [DevSecOps controls](#).

- [Plan and develop](#): Bring security to the planning phase in modern development methodologies to implement threat modeling, IDE security plugins/pre-commit, and peer review.
- [Commit the code](#): Evaluate and implement vulnerability scanning capability to your centralized repositories to discover risks and perform remediation.
- [Build and test](#): Use build and release pipelines for automation and standardization for the processes of building and deploying secure code without spending large amounts of time in redeploying or upgrading existing environments.
- [Go to production and operate](#): Oversee and manage the state of security when the solution is brought to production. Use infrastructure scanning tools and penetration testing practices for enabling teams to find risks and vulnerabilities to address.

Test-driven development cycle

Before beginning any security improvements, it's important to understand the "definition of done" and all "acceptance criteria". For more information, see the articles on [test-driven development of landing zones](#) and [test-driven development in Azure](#).

Next steps

Understand how to [improve landing zone operations](#) to support critical applications.

[Improve landing zone operations](#)

Connect your Azure subscriptions

Article • 08/07/2024

In this guide, you'll learn how to enable Microsoft Defender for Cloud on your Azure subscription.

Microsoft Defender for Cloud is a cloud-native application protection platform (CNAPP) with a set of security measures and practices designed to protect your cloud-based applications end-to-end by combining the following capabilities:

- A development security operations (DevSecOps) solution that unifies security management at the code level across multicloud and multiple-pipeline environments
- A cloud security posture management (CSPM) solution that surfaces actions that you can take to prevent breaches
- A cloud workload protection platform (CWPP) with specific protections for servers, containers, storage, databases, and other workloads

Defender for Cloud includes Foundational CSPM capabilities and access to [Microsoft Defender XDR](#) for free. You can add additional paid plans to secure all aspects of your cloud resources. You can try Defender for Cloud for free for the first 30 days. After 30 days, charges begin in accordance with the plans enabled in your environment. To learn more about these plans and their costs, see the Defender for Cloud [pricing page](#).

Important

Malware scanning in Defender for Storage is not included for free in the first 30 day trial and will be charged from the first day in accordance with the pricing scheme available on the Defender for Cloud [pricing page](#).

Defender for Cloud helps you find and fix security vulnerabilities. Defender for Cloud also applies access and application controls to block malicious activity, detect threats using analytics and intelligence, and respond quickly when under attack.

Prerequisites

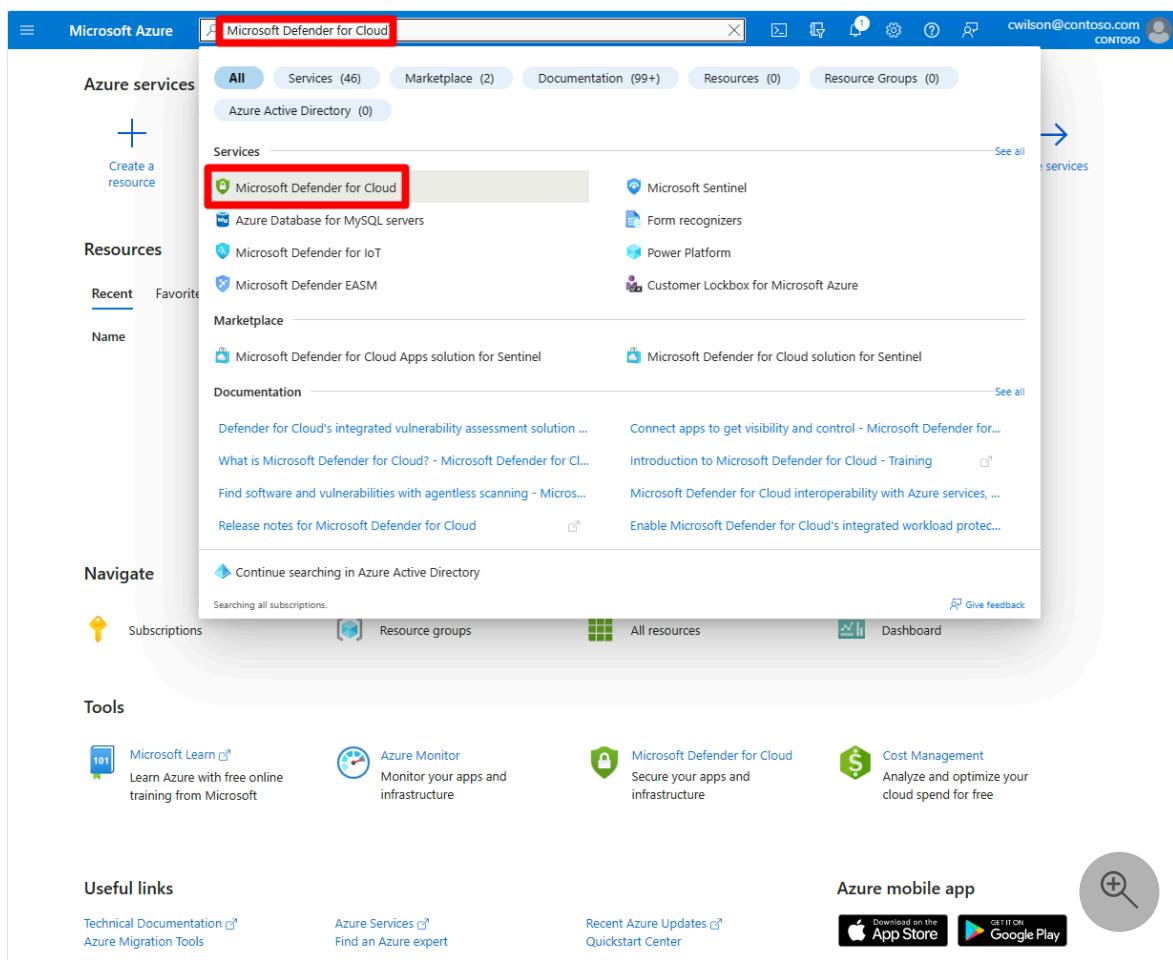
- To view information related to a resource in Defender for Cloud, you must be assigned the Owner, Contributor, or Reader role for the subscription or for the resource group that the resource is located in.

Enable Defender for Cloud on your Azure subscription

Tip

To enable Defender for Cloud on all subscriptions within a management group, see [Enable Defender for Cloud on multiple Azure subscriptions](#).

1. Sign in to the [Azure portal](#).
2. Search for and select Microsoft Defender for Cloud.



The Defender for Cloud's overview page opens.

Defender for Cloud is now enabled on your subscription and you have access to the basic features provided by Defender for Cloud. These features include:

- The [Foundational Cloud Security Posture Management \(CSPM\) plan](#).
- [Recommendations](#).
- Access to the [Asset inventory](#).
- [Workbooks](#).
- [Secure score](#).
- [Regulatory compliance](#) with the [Microsoft cloud security benchmark](#).

The Defender for Cloud overview page provides a unified view into the security posture of your hybrid cloud workloads, helping you discover and assess the security of your workloads and to identify and mitigate risks. Learn more in [Microsoft Defender for Cloud's overview page](#).

You can view and filter your list of subscriptions from the subscriptions menu to have Defender for Cloud adjust the overview page display to reflect the security posture to the selected subscriptions.

Within minutes of launching Defender for Cloud for the first time, you might see:

- [Recommendations](#) for ways to improve the security of your connected resources.
- An inventory of your resources that Defender for Cloud assesses along with the security posture of each.

Enable all paid plans on your subscription

To enable all of Defender for Cloud's protections, you need to enable the plans for the workloads that you want to protect.

Note

- You can enable **Microsoft Defender for Storage accounts**, **Microsoft Defender for SQL**, **Microsoft Defender for open-source relational databases** at either the subscription level or resource level.
- The Microsoft Defender plans available at the workspace level are: **Microsoft Defender for Servers**, **Microsoft Defender for SQL servers on machines**.

Important

Microsoft Defender for SQL is a subscription level bundle that uses either a default or custom workspace.

When you enable Defender plans on an entire Azure subscription, the protections are applied to all other resources in the subscription.

To enable additional paid plans on a subscription:

1. Sign in to the [Azure portal](#).
2. Search for and select **Microsoft Defender for Cloud**.
3. In the Defender for Cloud menu, select **Environment settings**.

The screenshot shows the Microsoft Defender for Cloud portal's General dashboard. On the left, there's a vertical navigation menu with sections: General, Cloud Security, and Management. Under General, the 'Environment settings' option is highlighted with a red box. The Cloud Security section includes links for Security posture, Regulatory compliance, Workload protections, Firewall Manager, and DevOps Security (Preview). The Management section includes links for Environment settings, Security solutions, and Workflow automation. To the right of the menu, there are several summary cards: 'Defer Click' (with a blue info icon), '93 Azure subs' (with a yellow key icon), '323 Active rec' (with a blue list icon), a shield icon with a star, '15 Unassign recom' (with a blue people icon), and a large circular progress bar labeled 'Secure s'. At the bottom right, there's a blue 'Explore' button.

- General
- Overview
- Getting started
- Recommendations
- Security alerts
- Inventory
- Cloud Security Explorer (Preview)
- Workbooks
- Community
- Diagnose and solve problems

- Cloud Security
- Security posture
- Regulatory compliance
- Workload protections
- Firewall Manager
- DevOps Security (Preview)

- Management
- Environment settings
- Security solutions
- Workflow automation

4. Select the subscription or workspace that you want to protect.
5. Select **Enable all** to enable all of the plans for Defender for Cloud.

6. Select Save.

All of the plans are turned on and the monitoring components required by each plan are deployed to the protected resources.

If you want to disable any of the plans, toggle the individual plan to **off**. The extensions used by the plan aren't uninstalled but, after a short time, the extensions stop collecting data.

Tip

To enable Defender for Cloud on all subscriptions within a management group, see [Enable Defender for Cloud on multiple Azure subscriptions](#).

Integrate with Microsoft Defender XDR

When you enable Defender for Cloud, Defender for Cloud's alerts are automatically integrated into the Microsoft Defender Portal. No further steps are needed.

The integration between Microsoft Defender for Cloud and Microsoft Defender XDR brings your cloud environments into Microsoft Defender XDR. With Defender for Cloud's alerts and cloud correlations integrated into Microsoft Defender XDR, SOC teams can now access all security information from a single interface.

Learn more about Defender for Cloud's [alerts in Microsoft Defender XDR](#).

Next steps

In this guide, you enabled Defender for Cloud on your Azure subscription. The next step is to set up your hybrid and multicloud environments.

Quickstart: Connect your non-Azure machines to Microsoft Defender for Cloud with Azure Arc

Quickstart: Connect your AWS accounts to Microsoft Defender for Cloud

Quickstart: Connect your GCP projects to Microsoft Defender for Cloud

Quickstart: Connect your non-Azure machines to Microsoft Defender for Cloud with Defender for Endpoint

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Ask the community](#)

Quickstart: Onboard Microsoft Sentinel

Article • 06/18/2024

In this quickstart, you'll enable Microsoft Sentinel and install a solution from the content hub. Then, you'll set up a data connector to start ingesting data into Microsoft Sentinel.

Microsoft Sentinel comes with many data connectors for Microsoft products such as the Microsoft Defender XDR service-to-service connector. You can also enable built-in connectors for non-Microsoft products such as Syslog or Common Event Format (CEF). For this quickstart, you'll use the Azure Activity data connector that's available in the Azure Activity solution for Microsoft Sentinel.

To onboard to Microsoft Sentinel by using the API, see the latest supported version of [Sentinel Onboarding States](#).

Prerequisites

- **Active Azure Subscription.** If you don't have one, create a [free account](#) before you begin.
- **Log Analytics workspace.** Learn how to [create a Log Analytics workspace](#). For more information about Log Analytics workspaces, see [Designing your Azure Monitor Logs deployment](#).

You may have a default of [30 days retention](#) in the Log Analytics workspace used for Microsoft Sentinel. To make sure that you can use all Microsoft Sentinel functionality and features, raise the retention to 90 days. [Configure data retention and archive policies in Azure Monitor Logs](#).

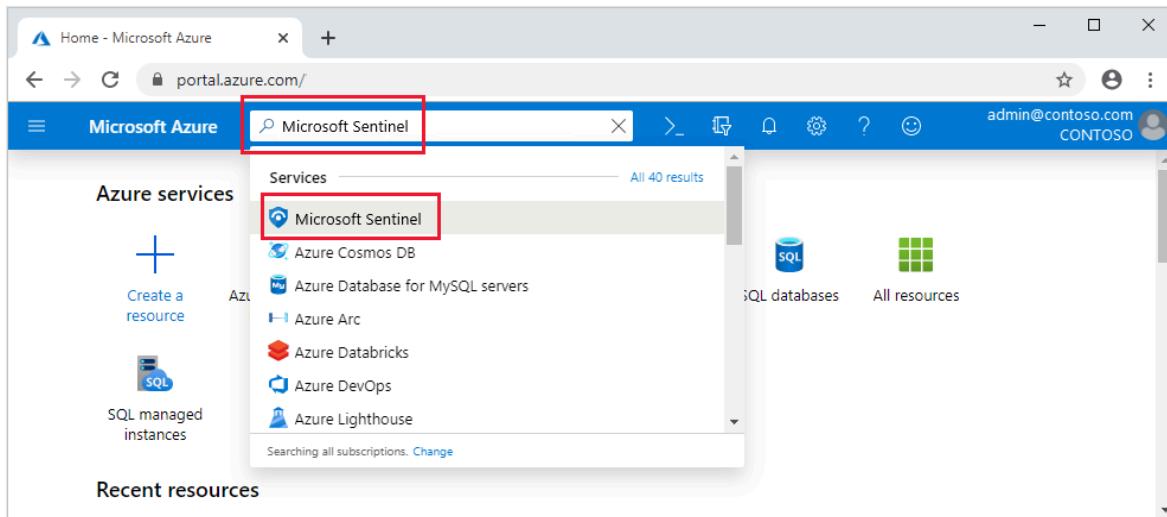
- **Permissions:**
 - To enable Microsoft Sentinel, you need **contributor** permissions to the subscription in which the Microsoft Sentinel workspace resides.
 - To use Microsoft Sentinel, you need either **Microsoft Sentinel Contributor** or **Microsoft Sentinel Reader** permissions on the resource group that the workspace belongs to.
 - To install or manage solutions in the content hub, you need the **Microsoft Sentinel Contributor** role on the resource group that the workspace belongs to.
- **Microsoft Sentinel is a paid service.** Review the [pricing options](#) and the [Microsoft Sentinel pricing page](#).

- Before deploying Microsoft Sentinel to a production environment, review the [predeployment activities and prerequisites for deploying Microsoft Sentinel](#).

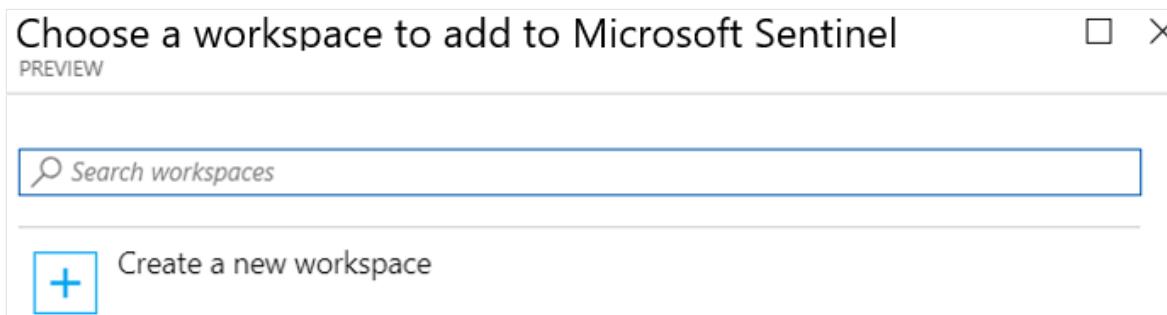
Enable Microsoft Sentinel

To get started, add Microsoft Sentinel to an existing workspace or create a new one.

1. Sign in to the [Azure portal](#).
2. Search for and select **Microsoft Sentinel**.



3. Select **Create**.
4. Select the workspace you want to use or create a new one. You can run Microsoft Sentinel on more than one workspace, but the data is isolated to a single workspace.



- The default workspaces created by Microsoft Defender for Cloud aren't shown in the list. You can't install Microsoft Sentinel on these workspaces.
- Once deployed on a workspace, Microsoft Sentinel **doesn't support** moving that workspace to another resource group or subscription.

5. Select **Add**.

Install a solution from the content hub

The content hub in Microsoft Sentinel is the centralized location to discover and manage out-of-the-box content including data connectors. For this quickstart, install the solution for Azure Activity.

1. In Microsoft Sentinel, select **Content hub**.
2. Find and select the **Azure Activity** solution.

The screenshot shows the Microsoft Sentinel Content hub interface. On the left, there's a sidebar with navigation links: General (Overview (Preview), Logs, News & guides, Search), Threat management (Incidents, Workbooks, Hunting, Notebooks, Entity behavior, Threat intelligence, MITRE ATT&CK (Preview)), and Content management (Content hub, Repositories (Preview)). The main area displays statistics: 290 Solutions, 246 Standalone contents, 6 Installed, and 0 Updates. A search bar at the top right contains 'azure activity'. Below it are filters for Support: All, Provider: All, Category: All, and Content sources: All. A table lists the solutions, with 'Azure Activity' highlighted as 'FEATURED'. The table columns are Content title, Content source, and Provider. The Azure Activity row shows 'Azure Activity' under Content title, 'Solution' under Content source, and 'Microsoft' under Provider.

Content title	Content source	Provider
Azure Activity FEATURED	Solution	Microsoft
Azure CloudShell Usage	Standalone	
Azure SensitiveOperations Review Workb	Standalone	

3. On the toolbar at the top of the page, select **Install/Update**.

Set up the data connector

Microsoft Sentinel ingests data from services and apps by connecting to the service and forwarding the events and logs to Microsoft Sentinel. For this quickstart, install the data connector to forward data for Azure Activity to Microsoft Sentinel.

1. In Microsoft Sentinel, select **Data connectors**.
2. Search for and select the **Azure Activity** data connector.
3. In the details pane for the connector, select **Open connector page**.

4. Review the instructions to configure the connector.
5. Select **Launch Azure Policy Assignment Wizard**.
6. On the **Basics** tab, set the **Scope** to the subscription and resource group that has activity to send to Microsoft Sentinel. For example, select the subscription that contains your Microsoft Sentinel instance.
7. Select the **Parameters** tab.
8. Set the **Primary Log Analytics workspace**. This should be the workspace where Microsoft Sentinel is installed.
9. Select **Review + create** and **Create**.

Generate activity data

Let's generate some activity data by enabling a rule that was included in the Azure Activity solution for Microsoft Sentinel. This step also shows you how to manage content in the content hub.

1. In Microsoft Sentinel, select **Content hub**.
2. Find and select the **Azure Activity** solution.
3. From the right-hand side pane, select **Manage**.
4. Find and select the rule template **Suspicious Resource deployment**.
5. Select **Configuration**.
6. Select the rule and **Create rule**.
7. On the **General** tab, change the **Status** to enabled. Leave the rest of the default values.
8. Accept the defaults on the other tabs.
9. On the **Review and create** tab, select **Create**.

View data ingested into Microsoft Sentinel

Now that you've enabled the Azure Activity data connector and generated some activity data let's view the activity data added to the workspace.

1. In Microsoft Sentinel, select **Data connectors**.
2. Search for and select the **Azure Activity** data connector.
3. In the details pane for the connector, select **Open connector page**.
4. Review the **Status** of the data connector. It should be **Connected**.

Azure Activity

«



Azure Activity

Connected

Status

Microsoft

Provider

14 minutes ago

Last Log Received

Description

Azure Activity Log is a subscription log that provides insight into subscription-level events that occur in Azure, including events from Azure Resource Manager operational data, service health events, write operations taken on the resources in your subscription, and the status of activities performed in Azure.

Last data received

06/21/23, 02:29 PM

Content source

Azure Activity

Version

2.0.0

Author

Microsoft

Supported by

[Microsoft Corporation](#) |

[Email](#)

Related content

4
Workbooks

2
Queries

25
Analytics rules templates

Data received

200

[Go to log analytics](#)

5. In the left-hand side pane above the chart, select [Go to log analytics](#).

6. On the top of the pane, next to the **New query 1** tab, select the + to add a new query tab.

7. In the query pane, run the following query to view the activity data ingested into the workspace.

The screenshot shows the Microsoft Sentinel Log Analytics workspace. At the top, there's a Kusto query editor with the text "Kusto" and a "AzureActivity" query. Below the editor is a breadcrumb navigation bar: Home > Microsoft Sentinel > Microsoft Sentinel | Data connectors > Azure Activity >. The main area is titled "Logs" with a "Tables" tab selected. There are two open queries: "New Query 1*" and "New Query 2*". The "Run" button is highlighted. The time range is set to "Last 24 hours". Below the queries is a table named "AzureActivity". The table has two columns: "TimeGenerated [UTC]" and "OperationNameValue". The data in the table is as follows:

TimeGenerated [UTC]	OperationNameValue
6/21/2023, 9:39:36.954 PM	MICROSOFT.SECURITYINSIGHTS/CONTENTPACKAGES/DELE...
6/21/2023, 9:39:37.407 PM	MICROSOFT.SECURITYINSIGHTS/CONTENTPACKAGES/DELE...
6/21/2023, 9:40:11.777 PM	MICROSOFT.SECURITYINSIGHTS/CONTENTTEMPLATES/WRI...
6/21/2023, 9:40:11.949 PM	MICROSOFT.SECURITYINSIGHTS/CONTENTTEMPLATES/WRI...
6/21/2023, 9:40:11.855 PM	MICROSOFT.SECURITYINSIGHTS/CONTENTTEMPLATES/WRI...
6/21/2023, 9:40:12.105 PM	MICROSOFT.SECURITYINSIGHTS/CONTENTTEMPLATES/WRI...
6/21/2023, 9:40:10.714 PM	MICROSOFT.RESOURCES/DEPLOYMENTS/WRITE
6/21/2023, 9:40:11.002 PM	MICROSOFT.DOCUMENTS/DEPLOYMENTS/audit

Next steps

In this quickstart, you enabled Microsoft Sentinel and installed a solution from the content hub. Then, you set up a data connector to start ingesting data into Microsoft Sentinel. You also verified that data is being ingested by viewing the data in the workspace.

- To visualize the data you've collected by using the dashboards and workbooks, see [Visualize collected data](#).
- To detect threats by using analytics rules, see [Tutorial: Detect threats by using analytics rules in Microsoft Sentinel](#).

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback ↗

Implement a secure hybrid network

Azure Firewall

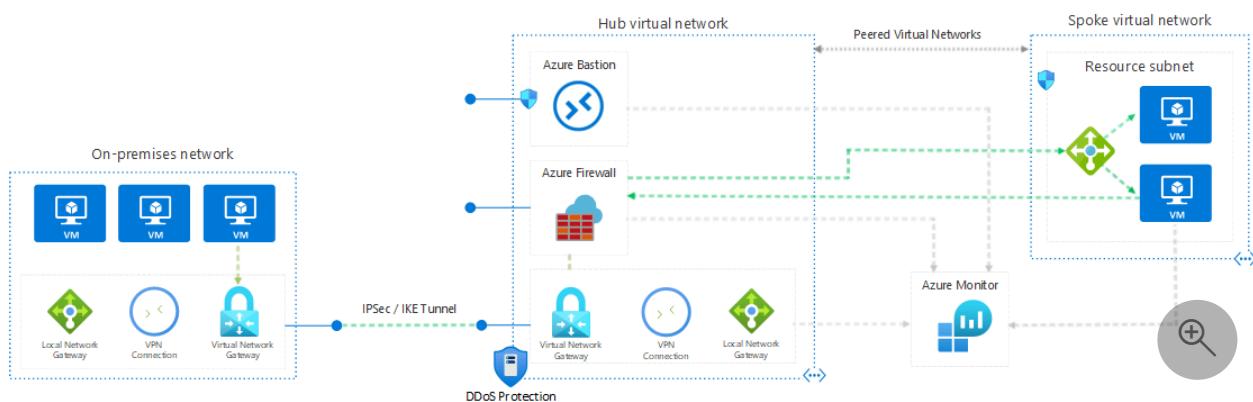
Azure Load Balancer

Azure Virtual Machines

Azure Virtual Network

This reference architecture shows a secure hybrid network that extends an on-premises network to Azure. The architecture implements a *perimeter network*, also called a *DMZ*, between the on-premises network and an Azure virtual network. All inbound and outbound traffic passes through Azure Firewall.

Architecture



Download a [Visio file](#) of this architecture.

Components

The architecture consists of the following aspects:

- **On-premises network.** A private local-area network implemented in an organization.
- **Azure virtual network.** The virtual network hosts the solution components and other resources running in Azure.

[Virtual network routes](#) define the flow of IP traffic within the Azure virtual network. In the diagram, there are two user-defined route tables.

In the gateway subnet, traffic is routed through the Azure Firewall instance.

! Note

Depending on the requirements of your VPN connection, you can configure Border Gateway Protocol (BGP) routes to implement the forwarding rules that direct traffic back through the on-premises network.

- **Gateway.** The gateway provides connectivity between the routers in the on-premises network and the virtual network. The gateway is placed in its own subnet.
- **Azure Firewall.** [Azure Firewall](#) is a managed firewall as a service. The Firewall instance is placed in its own subnet.
- **Network security groups.** Use [security groups](#) to restrict network traffic within the virtual network.
- **Azure Bastion.** [Azure Bastion](#) allows you to log into virtual machines (VMs) in the virtual network through SSH or remote desktop protocol (RDP) without exposing the VMs directly to the internet. Use Bastion to manage the VMs in the virtual network.

Bastion requires a dedicated subnet named [AzureBastionSubnet](#).

Potential use cases

This architecture requires a connection to your on-premises datacenter, using either a [VPN gateway](#) or an ExpressRoute connection. Typical uses for this architecture include:

- Hybrid applications where workloads run partly on-premises and partly in Azure.
- Infrastructure that requires granular control over traffic entering an Azure virtual network from an on-premises datacenter.
- Applications that must audit outgoing traffic. Auditing is often a regulatory requirement of many commercial systems and can help to prevent public disclosure of private information.

Recommendations

The following recommendations apply for most scenarios. Follow these recommendations unless you have a specific requirement that overrides them.

Access control recommendations

Use [Azure role-based access control \(Azure RBAC\)](#) to manage the resources in your application. Consider creating the following [custom roles](#):

- A DevOps role with permissions to administer the infrastructure for the application, deploy the application components, and monitor and restart VMs.
- A centralized IT administrator role to manage and monitor network resources.
- A security IT administrator role to manage secure network resources such as the firewall.

The IT administrator role shouldn't have access to the firewall resources. Access should be restricted to the security IT administrator role.

Resource group recommendations

Azure resources such as VMs, virtual networks, and load balancers can be easily managed by grouping them together into resource groups. Assign Azure roles to each resource group to restrict access.

We recommend creating the following resource groups:

- A resource group containing the virtual network (excluding the VMs), NSGs, and the gateway resources for connecting to the on-premises network. Assign the centralized IT administrator role to this resource group.
- A resource group containing the VMs for the Azure Firewall instance and the user-defined routes for the gateway subnet. Assign the security IT administrator role to this resource group.
- Separate resource groups for each spoke virtual network that contains the load balancer and VMs.

Networking recommendations

To accept inbound traffic from the internet, add a [Destination Network Address Translation \(DNAT\)](#) rule to Azure Firewall.

- Destination address = Public IP address of the firewall instance.
- Translated address = Private IP address within the virtual network.

[Force-tunnel](#) all outbound internet traffic through your on-premises network using the site-to-site VPN tunnel, and route to the internet using network address translation (NAT). This design prevents accidental leakage of any confidential information and allows inspection and auditing of all outgoing traffic.

Don't completely block internet traffic from the resources in the spoke network subnets. Blocking traffic will prevent these resources from using Azure PaaS services that rely on

public IP addresses, such as VM diagnostics logging, downloading of VM extensions, and other functionality. Azure diagnostics also requires that components can read and write to an Azure Storage account.

Verify that outbound internet traffic is force-tunneled correctly. If you're using a VPN connection with the [routing and remote access service](#) on an on-premises server, use a tool such as [WireShark](#).

Consider using Application Gateway or Azure Front Door for SSL termination.

Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

For details about the bandwidth limits of VPN Gateway, see [Gateway SKUs](#). For higher bandwidths, consider upgrading to an ExpressRoute gateway. ExpressRoute provides up to 10-Gbps bandwidth with lower latency than a VPN connection.

For more information about the scalability of Azure gateways, see the scalability consideration sections in:

- [Implementing a hybrid network architecture with Azure and on-premises VPN](#)
- [Implementing a hybrid network architecture with Azure ExpressRoute](#)

For details about managing virtual networks and NSGs at scale, see [Azure Virtual Network Manager \(AVNM\): Create a secured hub and spoke network](#) to create new (and onboard existing) hub and spoke virtual network topologies for central management of connectivity and NSG rules.

Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

If you're using Azure ExpressRoute to provide connectivity between the virtual network and on-premises network, [configure a VPN gateway to provide failover](#) if the ExpressRoute connection becomes unavailable.

For information on maintaining availability for VPN and ExpressRoute connections, see the availability considerations in:

- [Implementing a hybrid network architecture with Azure and on-premises VPN](#)
- [Implementing a hybrid network architecture with Azure ExpressRoute](#)

Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

If gateway connectivity from your on-premises network to Azure is down, you can still reach the VMs in the Azure virtual network through Azure Bastion.

Each tier's subnet in the reference architecture is protected by NSG rules. You may need to create a rule to open port 3389 for remote desktop protocol (RDP) access on Windows VMs or port 22 for secure shell (SSH) access on Linux VMs. Other management and monitoring tools may require rules to open additional ports.

If you're using ExpressRoute to provide the connectivity between your on-premises datacenter and Azure, use the [Azure Connectivity Toolkit \(AzureCT\)](#) to monitor and troubleshoot connection issues.

You can find additional information about monitoring and managing VPN and ExpressRoute connections in the article [Implementing a hybrid network architecture with Azure and on-premises VPN](#).

Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

This reference architecture implements multiple levels of security.

Routing all on-premises user requests through Azure Firewall

The user-defined route in the gateway subnet blocks all user requests other than those received from on-premises. The route passes allowed requests to the firewall. The

requests are passed on to the resources in the spoke virtual networks if they're allowed by the firewall rules. You can add other routes, but make sure they don't inadvertently bypass the firewall or block administrative traffic intended for the management subnet.

Using NSGs to block/pass traffic to spoke virtual network subnets

Traffic to and from resource subnets in spoke virtual networks is restricted by using NSGs. If you have a requirement to expand the NSG rules to allow broader access to these resources, weigh these requirements against the security risks. Each new inbound pathway represents an opportunity for accidental or purposeful data leakage or application damage.

DDoS protection

Azure DDoS Protection, combined with application-design best practices, provides enhanced DDoS mitigation features to provide more defense against DDoS attacks. You should enable [Azure DDOS Protection](#) on any perimeter virtual network.

Use AVNM to create baseline Security Admin rules

AVNM allows you to create baselines of security rules, which can take priority over network security group rules. [Security admin rules](#) are evaluated before NSG rules and have the same nature of NSGs, with support for prioritization, service tags, and L3-L4 protocols. AVNM allows central IT to enforce a baseline of security rules, while allowing an independency of additional NSG rules by the spoke virtual network owners. To facilitate a controlled rollout of security rules changes, AVNM's [deployments](#) feature allows you to safely release of these configurations' breaking changes to the hub-and-spoke environments.

DevOps access

Use [Azure RBAC](#) to restrict the operations that DevOps can perform on each tier. When granting permissions, use the [principle of least privilege](#). Log all administrative operations and perform regular audits to ensure any configuration changes were planned.

Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost](#)

optimization pillar.

Use the [Azure pricing calculator](#) to estimate costs. Other considerations are described in the Cost optimization section in [Microsoft Azure Well-Architected Framework](#).

Here are cost considerations for the services used in this architecture.

Azure Firewall

In this architecture, Azure Firewall is deployed in the virtual network to control traffic between the gateway's subnet and the resources in the spoke virtual networks. In this way Azure Firewall is cost effective because it's used as a shared solution consumed by multiple workloads. Here are the Azure Firewall pricing models:

- Fixed rate per deployment hour.
- Data processed per GB to support auto scaling.

When compared to network virtual appliances (NVAs), with Azure Firewall you can save up to 30-50%. For more information, see [Azure Firewall vs NVA](#).

Azure Bastion

Azure Bastion securely connects to your virtual machine over RDP and SSH without having the need to configure a public IP on the virtual machine.

Bastion billing is comparable to a basic, low-level virtual machine configured as a jump box. Bastion is more cost effective than a jump box as it has built-in security features, and doesn't incur extra costs for storage and managing a separate server.

Azure Virtual Network

Azure Virtual Network is free. Every subscription is allowed to create up to 1,000 virtual networks across all regions. All traffic that occurs within the boundaries of a virtual network is free. For example, VMs in the same virtual network that talk to each other don't incur network traffic charges.

Internal load balancer

Basic load balancing between virtual machines that reside in the same virtual network is free.

In this architecture, internal load balancers are used to load balance traffic inside a virtual network.

Deploy this scenario

This deployment creates two resource groups; the first holds a mock on-premises network, the second a set of hub and spoke networks. The mock on-premises network and the hub network are connected using Azure Virtual Network gateways to form a site-to-site connection. This configuration is very similar to how you would connect your on-premises datacenter to Azure.

This deployment can take up to 45 minutes to complete. The recommended deployment method is using the portal option found below.

Azure portal

Use the following button to deploy the reference using the Azure portal.

 Deploy to Azure

Once the deployment has been completed, verify site-to-site connectivity by looking at the newly created connection resources. While in the Azure portal, search for 'connections' and note that the status of each connection.

Connections ⚙ ...

Microsoft

+ Add Edit columns Refresh | Assign tags

Subscriptions: 1 of 41 selected – Don't see a subscription? Open Directory + Subscription settings

Filter by name...	All resource groups	All locations	All tags		
2 items					
Name ↑	Status	Peer 1	Peer 2	Resource group ↑↓	Location ↑↓
<input type="checkbox"/> hub-to-mock-prem	Connected	vpn-azure-network	local-gateway-azure-netw...	site-to-site-azure-network	East US
<input type="checkbox"/> mock-prem-to-hub	Connected	vpn-mock-prem	local-gateway-moc-prem	site-to-site-mock-prem	East US

The IIS instance found in the spoke network can be accessed from the virtual machine located in the mock on-premises network. Create a connection to the virtual machine using the included Azure Bastion host, open a web browser, and navigate to the address of the application's network load balancer.

For detailed information and additional deployment options, see the Azure Resource Manager templates (ARM templates) used to deploy this solution: [Secure Hybrid Network](#).

Next steps

- [The virtual datacenter: A network perspective.](#)

- Azure security documentation.

Related resources

- Connect an on-premises network to Azure using ExpressRoute.
 - Configure ExpressRoute and Site-to-Site coexisting connections using PowerShell
 - Extend an on-premises network using ExpressRoute.
-

Feedback

Was this page helpful?



Yes



No

Azure Identity Management and access control security best practices

Article • 09/29/2024

In this article, we discuss a collection of Azure identity management and access control security best practices. These best practices are derived from our experience with [Microsoft Entra ID](#) and the experiences of customers like yourself.

For each best practice, we explain:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- Possible alternatives to the best practice
- How you can learn to enable the best practice

This Azure identity management and access control security best practices article is based on a consensus opinion and Azure platform capabilities and feature sets, as they exist at the time this article was written.

The intention in writing this article is to provide a general roadmap to a more robust security posture after deployment guided by our "[5 steps to securing your identity infrastructure](#)" checklist, which walks you through some of our core features and services.

Opinions and technologies change over time and this article will be updated on a regular basis to reflect those changes.

Azure identity management and access control security best practices discussed in this article include:

- Treat identity as the primary security perimeter
- Centralize identity management
- Manage connected tenants
- Enable single sign-on
- Turn on Conditional Access
- Plan for routine security improvements
- Enable password management
- Enforce multifactor verification for users
- Use role-based access control
- Lower exposure of privileged accounts

- Control locations where resources are located
- Use Microsoft Entra ID for storage authentication

Treat identity as the primary security perimeter

Many consider identity to be the primary perimeter for security. This is a shift from the traditional focus on network security. Network perimeters keep getting more porous, and that perimeter defense can't be as effective as it was before the explosion of [BYOD](#) devices and cloud applications.

[Microsoft Entra ID](#) is the Azure solution for identity and access management. Microsoft Entra ID is a multitenant, cloud-based directory and identity management service from Microsoft. It combines core directory services, application access management, and identity protection into a single solution.

The following sections list best practices for identity and access security using Microsoft Entra ID.

Best practice: Center security controls and detections around user and service identities.

Detail: Use Microsoft Entra ID to collocate controls and identities.

Centralize identity management

In a hybrid identity scenario we recommend that you integrate your on-premises and cloud directories. Integration enables your IT team to manage accounts from one location, regardless of where an account is created. Integration also helps your users be more productive by providing a common identity for accessing both cloud and on-premises resources.

Best practice: Establish a single Microsoft Entra instance. Consistency and a single authoritative source will increase clarity and reduce security risks from human errors and configuration complexity.

Detail: Designate a single Microsoft Entra directory as the authoritative source for corporate and organizational accounts.

Best practice: Integrate your on-premises directories with Microsoft Entra ID.

Detail: Use [Microsoft Entra Connect](#) to synchronize your on-premises directory with your cloud directory.

 Note

There are [factors that affect the performance of Microsoft Entra Connect](#). Ensure Microsoft Entra Connect has enough capacity to keep underperforming systems from impeding security and productivity. Large or complex organizations (organizations provisioning more than 100,000 objects) should follow the [recommendations](#) to optimize their Microsoft Entra Connect implementation.

Best practice: Don't synchronize accounts to Microsoft Entra ID that have high privileges in your existing Active Directory instance.

Detail: Don't change the default [Microsoft Entra Connect configuration](#) that filters out these accounts. This configuration mitigates the risk of adversaries pivoting from cloud to on-premises assets (which could create a major incident).

Best practice: Turn on password hash synchronization.

Detail: Password hash synchronization is a feature used to synch user password hashes from an on-premises Active Directory instance to a cloud-based Microsoft Entra instance. This sync helps to protect against leaked credentials being replayed from previous attacks.

Even if you decide to use federation with Active Directory Federation Services (AD FS) or other identity providers, you can optionally set up password hash synchronization as a backup in case your on-premises servers fail or become temporarily unavailable. This sync enables users to sign in to the service by using the same password that they use to sign in to their on-premises Active Directory instance. It also allows Identity Protection to detect compromised credentials by comparing synchronized password hashes with passwords known to be compromised, if a user has used the same email address and password on other services that aren't connected to Microsoft Entra ID.

For more information, see [Implement password hash synchronization with Microsoft Entra Connect Sync](#).

Best practice: For new application development, use Microsoft Entra ID for authentication.

Detail: Use the correct capabilities to support authentication:

- Microsoft Entra ID for employees
- [Microsoft Entra B2B](#) for guest users and external partners
- [Azure AD B2C](#) to control how customers sign up, sign in, and manage their profiles when they use your applications

Organizations that don't integrate their on-premises identity with their cloud identity can have more overhead in managing accounts. This overhead increases the likelihood of mistakes and security breaches.

Note

You need to choose which directories critical accounts will reside in and whether the admin workstation used is managed by new cloud services or existing processes. Using existing management and identity provisioning processes can decrease some risks but can also create the risk of an attacker compromising an on-premises account and pivoting to the cloud. You might want to use a different strategy for different roles (for example, IT admins vs. business unit admins). You have two options. First option is to create Microsoft Entra accounts that aren't synchronized with your on-premises Active Directory instance. Join your admin workstation to Microsoft Entra ID, which you can manage and patch by using Microsoft Intune. Second option is to use existing admin accounts by synchronizing to your on-premises Active Directory instance. Use existing workstations in your Active Directory domain for management and security.

Manage connected tenants

Your security organization needs visibility to assess risk and to determine whether the policies of your organization, and any regulatory requirements, are being followed. You should ensure that your security organization has visibility into all subscriptions connected to your production environment and network (via [Azure ExpressRoute](#) or [site-to-site VPN](#)). A [Global Administrator](#) in Microsoft Entra ID can elevate their access to the [User Access Administrator](#) role and see all subscriptions and managed groups connected to your environment.

See [elevate access to manage all Azure subscriptions and management groups](#) to ensure that you and your security group can view all subscriptions or management groups connected to your environment. You should remove this elevated access after you've assessed risks.

Enable single sign-on

In a mobile-first, cloud-first world, you want to enable single sign-on (SSO) to devices, apps, and services from anywhere so your users can be productive wherever and whenever. When you have multiple identity solutions to manage, this becomes an administrative problem not only for IT but also for users who have to remember multiple passwords.

By using the same identity solution for all your apps and resources, you can achieve SSO. And your users can use the same set of credentials to sign in and access the

resources that they need, whether the resources are located on-premises or in the cloud.

Best practice: Enable SSO.

Detail: Microsoft Entra ID [extends on-premises Active Directory](#) to the cloud. Users can use their primary work or school account for their domain-joined devices, company resources, and all of the web and SaaS applications that they need to get their jobs done. Users don't have to remember multiple sets of usernames and passwords, and their application access can be automatically provisioned (or deprovisioned) based on their organization group memberships and their status as an employee. And you can control that access for gallery apps or for your own on-premises apps that you've developed and published through the [Microsoft Entra application proxy](#).

Use SSO to enable users to access their [SaaS applications](#) based on their work or school account in Microsoft Entra ID. This is applicable not only for Microsoft SaaS apps, but also other apps, such as [Google Apps](#) and [Salesforce](#). You can configure your application to use Microsoft Entra ID as a [SAML-based identity provider](#). As a security control, Microsoft Entra ID does not issue a token that allows users to sign in to the application unless they have been granted access through Microsoft Entra ID. You can grant access directly, or through a group that users are a member of.

Organizations that don't create a common identity to establish SSO for their users and applications are more exposed to scenarios where users have multiple passwords. These scenarios increase the likelihood of users reusing passwords or using weak passwords.

Turn on Conditional Access

Users can access your organization's resources by using a variety of devices and apps from anywhere. As an IT admin, you want to make sure that these devices meet your standards for security and compliance. Just focusing on who can access a resource is not sufficient anymore.

To balance security and productivity, you need to think about how a resource is accessed before you can make a decision about access control. With Microsoft Entra Conditional Access, you can address this requirement. With Conditional Access, you can make automated access control decisions based on conditions for accessing your cloud apps.

Best practice: Manage and control access to corporate resources.

Detail: Configure common Microsoft Entra [Conditional Access policies](#) based on a group, location, and application sensitivity for SaaS apps and Microsoft Entra ID-connected apps.

Best practice: Block legacy authentication protocols.

Detail: Attackers exploit weaknesses in older protocols every day, particularly for password spray attacks. Configure Conditional Access to [block legacy protocols](#).

Plan for routine security improvements

Security is always evolving, and it is important to build into your cloud and identity management framework a way to regularly show growth and discover new ways to secure your environment.

Identity Secure Score is a set of recommended security controls that Microsoft publishes that works to provide you a numerical score to objectively measure your security posture and help plan future security improvements. You can also view your score in comparison to those in other industries as well as your own trends over time.

Best practice: Plan routine security reviews and improvements based on best practices in your industry.

Detail: Use the Identity Secure Score feature to rank your improvements over time.

Enable password management

If you have multiple tenants or you want to enable users to [reset their own passwords](#), it's important that you use appropriate security policies to prevent abuse.

Best practice: Set up self-service password reset (SSPR) for your users.

Detail: Use the Microsoft Entra ID [self-service password reset](#) feature.

Best practice: Monitor how or if SSPR is really being used.

Detail: Monitor the users who are registering by using the Microsoft Entra ID [Password Reset Registration Activity report](#). The reporting feature that Microsoft Entra ID provides helps you answer questions by using prebuilt reports. If you're appropriately licensed, you can also create custom queries.

Best practice: Extend cloud-based password policies to your on-premises infrastructure.

Detail: Enhance password policies in your organization by performing the same checks for on-premises password changes as you do for cloud-based password changes. Install [Microsoft Entra password protection](#) for Windows Server Active Directory agents on-premises to extend banned password lists to your existing infrastructure. Users and admins who change, set, or reset passwords on-premises are required to comply with the same password policy as cloud-only users.

Enforce multifactor verification for users

We recommend that you require two-step verification for all of your users. This includes administrators and others in your organization who can have a significant impact if their account is compromised (for example, financial officers).

There are multiple options for requiring two-step verification. The best option for you depends on your goals, the Microsoft Entra edition you're running, and your licensing program. See [How to require two-step verification for a user](#) to determine the best option for you. See the [Microsoft Entra ID](#) and [Microsoft Entra multifactor authentication](#) pricing pages for more information about licenses and pricing.

Following are options and benefits for enabling two-step verification:

Option 1: Enable MFA for all users and login methods with Microsoft Entra Security Defaults

Benefit: This option enables you to easily and quickly enforce MFA for all users in your environment with a stringent policy to:

- Challenge administrative accounts and administrative logon mechanisms
- Require MFA challenge via Microsoft Authenticator for all users
- Restrict legacy authentication protocols.

This method is available to all licensing tiers but is not able to be mixed with existing Conditional Access policies. You can find more information in [Microsoft Entra Security Defaults](#)

Option 2: Enable multifactor authentication by changing user state.

Benefit: This is the traditional method for requiring two-step verification. It works with both [Microsoft Entra multifactor authentication in the cloud](#) and [Azure Multi-Factor Authentication Server](#). Using this method requires users to perform two-step verification every time they sign in and overrides Conditional Access policies.

To determine where multifactor authentication needs to be enabled, see [Which version of Microsoft Entra multifactor authentication is right for my organization?](#).

Option 3: Enable multifactor authentication with Conditional Access policy.

Benefit: This option allows you to prompt for two-step verification under specific conditions by using [Conditional Access](#). Specific conditions can be user sign-in from different locations, untrusted devices, or applications that you consider risky. Defining specific conditions where you require two-step verification enables you to avoid constant prompting for your users, which can be an unpleasant user experience.

This is the most flexible way to enable two-step verification for your users. Enabling a Conditional Access policy works only for Microsoft Entra multifactor authentication in the cloud and is a premium feature of Microsoft Entra ID. You can find more information on this method in [Deploy cloud-based Microsoft Entra multifactor authentication](#).

Option 4: Enable multifactor authentication with Conditional Access policies by evaluating [Risk-based Conditional Access policies](#).

Benefit: This option enables you to:

- Detect potential vulnerabilities that affect your organization's identities.
- Configure automated responses to detected suspicious actions that are related to your organization's identities.
- Investigate suspicious incidents and take appropriate action to resolve them.

This method uses the Microsoft Entra ID Protection risk evaluation to determine if two-step verification is required based on user and sign-in risk for all cloud applications. This method requires Microsoft Entra ID P2 licensing. You can find more information on this method in [Microsoft Entra ID Protection](#).

 **Note**

Option 2, enabling multifactor authentication by changing the user state, overrides Conditional Access policies. Because options 3 and 4 use Conditional Access policies, you cannot use option 2 with them.

Organizations that don't add extra layers of identity protection, such as two-step verification, are more susceptible for credential theft attack. A credential theft attack can lead to data compromise.

Use role-based access control

Access management for cloud resources is critical for any organization that uses the cloud. [Azure role-based access control \(Azure RBAC\)](#) helps you manage who has access to Azure resources, what they can do with those resources, and what areas they have access to.

Designating groups or individual roles responsible for specific functions in Azure helps avoid confusion that can lead to human and automation errors that create security risks. Restricting access based on the [need to know](#) and [least privilege](#) security principles is imperative for organizations that want to enforce security policies for data access.

Your security team needs visibility into your Azure resources in order to assess and remediate risk. If the security team has operational responsibilities, they need additional permissions to do their jobs.

You can use [Azure RBAC](#) to assign permissions to users, groups, and applications at a certain scope. The scope of a role assignment can be a subscription, a resource group, or a single resource.

Best practice: Segregate duties within your team and grant only the amount of access to users that they need to perform their jobs. Instead of giving everybody unrestricted permissions in your Azure subscription or resources, allow only certain actions at a particular scope.

Detail: Use [Azure built-in roles](#) in Azure to assign privileges to users.

Note

Specific permissions create unneeded complexity and confusion, accumulating into a “legacy” configuration that’s difficult to fix without fear of breaking something. Avoid resource-specific permissions. Instead, use management groups for enterprise-wide permissions and resource groups for permissions within subscriptions. Avoid user-specific permissions. Instead, assign access to groups in Microsoft Entra ID.

Best practice: Grant security teams with Azure responsibilities access to see Azure resources so they can assess and remediate risk.

Detail: Grant security teams the Azure RBAC [Security Reader](#) role. You can use the root management group or the segment management group, depending on the scope of responsibilities:

- **Root management group** for teams responsible for all enterprise resources
- **Segment management group** for teams with limited scope (commonly because of regulatory or other organizational boundaries)

Best practice: Grant the appropriate permissions to security teams that have direct operational responsibilities.

Detail: Review the Azure built-in roles for the appropriate role assignment. If the built-in roles don’t meet the specific needs of your organization, you can create [Azure custom roles](#). As with built-in roles, you can assign custom roles to users, groups, and service principals at subscription, resource group, and resource scopes.

Best practices: Grant Microsoft Defender for Cloud access to security roles that need it. Defender for Cloud allows security teams to quickly identify and remediate risks.

Detail: Add security teams with these needs to the Azure RBAC [Security Admin](#) role so they can view security policies, view security states, edit security policies, view alerts and recommendations, and dismiss alerts and recommendations. You can do this by using the root management group or the segment management group, depending on the scope of responsibilities.

Organizations that don't enforce data access control by using capabilities like Azure RBAC might be giving more privileges than necessary to their users. This can lead to data compromise by allowing users to access types of data (for example, high business impact) that they shouldn't have.

Lower exposure of privileged accounts

Securing privileged access is a critical first step to protecting business assets. Minimizing the number of people who have access to secure information or resources reduces the chance of a malicious user getting access, or an authorized user inadvertently affecting a sensitive resource.

Privileged accounts are accounts that administer and manage IT systems. Cyber attackers target these accounts to gain access to an organization's data and systems. To secure privileged access, you should isolate the accounts and systems from the risk of being exposed to a malicious user.

We recommend that you develop and follow a roadmap to secure privileged access against cyber attackers. For information about creating a detailed roadmap to secure identities and access that are managed or reported in Microsoft Entra ID, Microsoft Azure, Microsoft 365, and other cloud services, review [Securing privileged access for hybrid and cloud deployments in Microsoft Entra ID](#).

The following summarizes the best practices found in [Securing privileged access for hybrid and cloud deployments in Microsoft Entra ID](#):

Best practice: Manage, control, and monitor access to privileged accounts.

Detail: Turn on [Microsoft Entra Privileged Identity Management](#). After you turn on Privileged Identity Management, you'll receive notification email messages for privileged access role changes. These notifications provide early warning when additional users are added to highly privileged roles in your directory.

Best practice: Ensure all critical admin accounts are managed Microsoft Entra accounts.

Detail: Remove any consumer accounts from critical admin roles (for example, Microsoft accounts like hotmail.com, live.com, and outlook.com).

Best practice: Ensure all critical admin roles have a separate account for administrative tasks in order to avoid phishing and other attacks to compromise administrative privileges.

Detail: Create a separate admin account that's assigned the privileges needed to perform the administrative tasks. Block the use of these administrative accounts for daily productivity tools like Microsoft 365 email or arbitrary web browsing.

Best practice: Identify and categorize accounts that are in highly privileged roles.

Detail: After turning on Microsoft Entra Privileged Identity Management, view the users who are in the global administrator, privileged role administrator, and other highly privileged roles. Remove any accounts that are no longer needed in those roles, and categorize the remaining accounts that are assigned to admin roles:

- Individually assigned to administrative users, and can be used for non-administrative purposes (for example, personal email)
- Individually assigned to administrative users and designated for administrative purposes only
- Shared across multiple users
- For emergency access scenarios
- For automated scripts
- For external users

Best practice: Implement "just in time" (JIT) access to further lower the exposure time of privileges and increase your visibility into the use of privileged accounts.

Detail: Microsoft Entra Privileged Identity Management lets you:

- Limit users to only taking on their privileges JIT.
- Assign roles for a shortened duration with confidence that the privileges are revoked automatically.

Best practice: Define at least two emergency access accounts.

Detail: Emergency access accounts help organizations restrict privileged access in an existing Microsoft Entra environment. These accounts are highly privileged and are not assigned to specific individuals. Emergency access accounts are limited to scenarios where normal administrative accounts can't be used. Organizations must limit the emergency account's usage to only the necessary amount of time.

Evaluate the accounts that are assigned or eligible for the global admin role. If you don't see any cloud-only accounts by using the *.onmicrosoft.com domain (intended for emergency access), create them. For more information, see [Managing emergency access administrative accounts in Microsoft Entra ID](#).

Best practice: Have a "break glass" process in place in case of an emergency.

Detail: Follow the steps in [Securing privileged access for hybrid and cloud deployments in Microsoft Entra ID](#).

Best practice: Require all critical admin accounts to be password-less (preferred), or require multifactor authentication.

Detail: Use the [Microsoft Authenticator app](#) to sign in to any Microsoft Entra account without using a password. Like [Windows Hello for Business](#), the Microsoft Authenticator uses key-based authentication to enable a user credential that's tied to a device and uses biometric authentication or a PIN.

Require Microsoft Entra multifactor authentication at sign-in for all individual users who are permanently assigned to one or more of the Microsoft Entra admin roles: Global Administrator, Privileged Role Administrator, Exchange Online Administrator, and SharePoint Online Administrator. Enable [multifactor authentication for your admin accounts](#) and ensure that admin account users have registered.

Best practice: For critical admin accounts, have an admin workstation where production tasks aren't allowed (for example, browsing and email). This will protect your admin accounts from attack vectors that use browsing and email and significantly lower your risk of a major incident.

Detail: Use an admin workstation. Choose a level of workstation security:

- Highly secure productivity devices provide advanced security for browsing and other productivity tasks.
- [Privileged Access Workstations \(PAWs\)](#) provide a dedicated operating system that's protected from internet attacks and threat vectors for sensitive tasks.

Best practice: Deprovision admin accounts when employees leave your organization.

Detail: Have a process in place that disables or deletes admin accounts when employees leave your organization.

Best practice: Regularly test admin accounts by using current attack techniques.

Detail: Use Microsoft 365 Attack Simulator or a third-party offering to run realistic attack scenarios in your organization. This can help you find vulnerable users before a real attack occurs.

Best practice: Take steps to mitigate the most frequently used attacked techniques.

Detail: [Identify Microsoft accounts in administrative roles that need to be switched to work or school accounts](#)

[Ensure separate user accounts and mail forwarding for global administrator accounts](#)

[Ensure that the passwords of administrative accounts have recently changed](#)

[Turn on password hash synchronization](#)

[Require multifactor authentication for users in all privileged roles as well as exposed users](#)

[Obtain your Microsoft 365 Secure Score \(if using Microsoft 365\)](#)

[Review the Microsoft 365 security guidance \(if using Microsoft 365\)](#)

[Configure Microsoft 365 Activity Monitoring \(if using Microsoft 365\)](#)

[Establish incident/emergency response plan owners](#)

[Secure on-premises privileged administrative accounts](#)

If you don't secure privileged access, you might find that you have too many users in highly privileged roles and are more vulnerable to attacks. Malicious actors, including cyber attackers, often target admin accounts and other elements of privileged access to gain access to sensitive data and systems by using credential theft.

Control locations where resources are created

Enabling cloud operators to perform tasks while preventing them from breaking conventions that are needed to manage your organization's resources is very important. Organizations that want to control the locations where resources are created should hard code these locations.

You can use [Azure Resource Manager](#) to create security policies whose definitions describe the actions or resources that are specifically denied. You assign those policy definitions at the desired scope, such as the subscription, the resource group, or an individual resource.

 **Note**

Security policies are not the same as Azure RBAC. They actually use Azure RBAC to authorize users to create those resources.

Organizations that are not controlling how resources are created are more susceptible to users who might abuse the service by creating more resources than they need. Hardening the resource creation process is an important step to securing a multitenant scenario.

Actively monitor for suspicious activities

An active identity monitoring system can quickly detect suspicious behavior and trigger an alert for further investigation. The following table lists Microsoft Entra capabilities that can help organizations monitor their identities:

Best practice: Have a method to identify:

- Attempts to sign in [without being traced](#).
- [Brute force](#) attacks against a particular account.
- Attempts to sign in from multiple locations.
- Sign-ins from [infected devices](#).
- Suspicious IP addresses.

Detail: Use Microsoft Entra ID P1 or P2 [anomaly reports](#). Have processes and procedures in place for IT admins to run these reports on a daily basis or on demand (usually in an incident response scenario).

Best practice: Have an active monitoring system that notifies you of risks and can adjust risk level (high, medium, or low) to your business requirements.

Detail: Use [Microsoft Entra ID Protection](#), which flags the current risks on its own dashboard and sends daily summary notifications via email. To help protect your organization's identities, you can configure risk-based policies that automatically respond to detected issues when a specified risk level is reached.

Organizations that don't actively monitor their identity systems are at risk of having user credentials compromised. Without knowledge that suspicious activities are taking place through these credentials, organizations can't mitigate this type of threat.

Use Microsoft Entra ID for storage authentication

[Azure Storage](#) supports authentication and authorization with Microsoft Entra ID for Blob storage and Queue storage. With Microsoft Entra authentication, you can use the Azure role-based access control to grant specific permissions to users, groups, and applications down to the scope of an individual blob container or queue.

We recommend that you use [Microsoft Entra ID for authenticating access to storage](#) ↗.

Next step

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Get help at Microsoft Q&A](#)

Azure best practices for network security

Article • 09/27/2024

This article discusses a collection of Azure best practices to enhance your network security. These best practices are derived from our experience with Azure networking and the experiences of customers like yourself.

For each best practice, this article explains:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- Possible alternatives to the best practice
- How you can learn to enable the best practice

These best practices are based on a consensus opinion, and Azure platform capabilities and feature sets, as they exist at the time this article was written. Opinions and technologies change over time and this article will be updated regularly to reflect those changes.

Use strong network controls

You can connect [Azure virtual machines \(VMs\)](#) and appliances to other networked devices by placing them on [Azure virtual networks](#). That is, you can connect virtual network interface cards to a virtual network to allow TCP/IP-based communications between network-enabled devices. Virtual machines connected to an Azure virtual network can connect to devices on the same virtual network, different virtual networks, the internet, or your own on-premises networks.

As you plan your network and the security of your network, we recommend that you centralize:

- Management of core network functions like ExpressRoute, virtual network and subnet provisioning, and IP addressing.
- Governance of network security elements, such as network virtual appliance functions like ExpressRoute, virtual network and subnet provisioning, and IP addressing.

If you use a common set of management tools to monitor your network and the security of your network, you get clear visibility into both. A straightforward, unified security

strategy reduces errors because it increases human understanding and the reliability of automation.

Logically segment subnets

Azure virtual networks are similar to LANs on your on-premises network. The idea behind an Azure virtual network is that you create a network, based on a single private IP address space, on which you can place all your Azure virtual machines. The private IP address spaces available are in the Class A (10.0.0.0/8), Class B (172.16.0.0/12), and Class C (192.168.0.0/16) ranges.

Best practices for logically segmenting subnets include:

Best practice: Don't assign allow rules with broad ranges (for example, allow 0.0.0.0 through 255.255.255.255).

Detail: Ensure troubleshooting procedures discourage or ban setting up these types of rules. These allow rules lead to a false sense of security and are frequently found and exploited by red teams.

Best practice: Segment the larger address space into subnets.

Detail: Use [CIDR ↗](#)-based subnetting principles to create your subnets.

Best practice: Create network access controls between subnets. Routing between subnets happens automatically, and you don't need to manually configure routing tables. By default, there are no network access controls between the subnets that you create on an Azure virtual network.

Detail: Use a [network security group](#) to protect against unsolicited traffic into Azure subnets. Network security groups (NSGs) are simple, stateful packet inspection devices. NSGs use the 5-tuple approach (source IP, source port, destination IP, destination port and protocol) to create allow/deny rules for network traffic. You allow or deny traffic to and from a single IP address, to and from multiple IP addresses, or to and from entire subnets.

When you use network security groups for network access control between subnets, you can put resources that belong to the same security zone or role in their own subnets.

Best practice: Avoid small virtual networks and subnets to ensure simplicity and flexibility. **Detail:** Most organizations add more resources than initially planned, and reallocating addresses is labor intensive. Using small subnets adds limited security value, and mapping a network security group to each subnet adds overhead. Define subnets broadly to ensure that you have flexibility for growth.

Best practice: Simplify network security group rule management by defining [Application Security Groups](#).

Detail: Define an Application Security Group for lists of IP addresses that you think might change in the future or be used across many network security groups. Be sure to name Application Security Groups clearly so others can understand their content and purpose.

Adopt a Zero Trust approach

Perimeter-based networks operate on the assumption that all systems within a network can be trusted. But today's employees access their organization's resources from anywhere on various devices and apps, which makes perimeter security controls irrelevant. Access control policies that focus only on who can access a resource aren't enough. To master the balance between security and productivity, security admins also need to factor in *how* a resource is being accessed.

Networks need to evolve from traditional defenses because networks might be vulnerable to breaches: an attacker can compromise a single endpoint within the trusted boundary and then quickly expand a foothold across the entire network. [Zero Trust](#) networks eliminate the concept of trust based on network location within a perimeter. Instead, Zero Trust architectures use device and user trust claims to gate access to organizational data and resources. For new initiatives, adopt Zero Trust approaches that validate trust at the time of access.

Best practices are:

Best practice: Give Conditional Access to resources based on device, identity, assurance, network location, and more.

Detail: [Microsoft Entra Conditional Access](#) lets you apply the right access controls by implementing automated access control decisions based on the required conditions. For more information, see [Manage access to Azure management with Conditional Access](#).

Best practice: Enable port access only after workflow approval.

Detail: You can use [just-in-time VM access in Microsoft Defender for Cloud](#) to lock down inbound traffic to your Azure VMs, reducing exposure to attacks while providing easy access to connect to VMs when needed.

Best practice: Grant temporary permissions to perform privileged tasks, which prevents malicious or unauthorized users from gaining access after the permissions have expired. Access is granted only when users need it.

Detail: Use just-in-time access in Microsoft Entra Privileged Identity Management or in a third-party solution to grant permissions to perform privileged tasks.

Zero Trust is the next evolution in network security. The state of cyberattacks drives organizations to take the "assume breach" mindset, but this approach shouldn't be limiting. Zero Trust networks protect corporate data and resources while ensuring that organizations can build a modern workplace by using technologies that empower employees to be productive anytime, anywhere, in any way.

Control routing behavior

When you put a virtual machine on an Azure virtual network, the VM can connect to any other VM on the same virtual network, even if the other VMs are on different subnets. This is possible because a collection of system routes enabled by default allows this type of communication. These default routes allow VMs on the same virtual network to initiate connections with each other, and with the internet (for outbound communications to the internet only).

Although the default system routes are useful for many deployment scenarios, there are times when you want to customize the routing configuration for your deployments. You can configure the next-hop address to reach specific destinations.

We recommend that you configure [user-defined routes](#) when you deploy a security appliance for a virtual network. We talk about this recommendation in a later section titled [secure your critical Azure service resources to only your virtual networks](#).

ⓘ Note

User-defined routes aren't required, and the default system routes usually work.

Use virtual network appliances

Network security groups and user-defined routing can provide a certain measure of network security at the network and transport layers of the [OSI model](#). But in some situations, you want or need to enable security at high levels of the stack. In such situations, we recommend that you deploy virtual network security appliances provided by Azure partners.

Azure network security appliances can deliver better security than what network-level controls provide. Network security capabilities of virtual network security appliances include:

- Firewalling
- Intrusion detection/intrusion prevention

- Vulnerability management
- Application control
- Network-based anomaly detection
- Web filtering
- Antivirus
- Botnet protection

To find available Azure virtual network security appliances, go to the [Azure Marketplace](#) and search for "security" and "network security."

Deploy perimeter networks for security zones

A [perimeter network](#) (also known as a DMZ) is a physical or logical network segment that provides an extra layer of security between your assets and the internet. Specialized network access control devices on the edge of a perimeter network allow only desired traffic into your virtual network.

Perimeter networks are useful because you can focus your network access control management, monitoring, logging, and reporting on the devices at the edge of your Azure virtual network. A perimeter network is where you typically enable [distributed denial of service \(DDoS\) protection](#), intrusion detection/intrusion prevention systems (IDS/IPS), firewall rules and policies, web filtering, network antimalware, and more. The network security devices sit between the internet and your Azure virtual network and have an interface on both networks.

Although this is the basic design of a perimeter network, there are many different designs, like back-to-back, tri-homed, and multi-homed.

Based on the Zero Trust concept mentioned earlier, we recommend that you consider using a perimeter network for all high security deployments to enhance the level of network security and access control for your Azure resources. You can use Azure or a third-party solution to provide an extra layer of security between your assets and the internet:

- Azure native controls. [Azure Firewall](#) and [Azure Web Application Firewall](#) offer basic security advantages. Advantages are a fully stateful firewall as a service, built-in high availability, unrestricted cloud scalability, FQDN filtering, support for OWASP core rule sets, and simple setup and configuration.
- Third-party offerings. Search the [Azure Marketplace](#) for next-generation firewall (NGFW) and other third-party offerings that provide familiar security tools and enhanced levels of network security. Configuration might be more complex, but a third-party offering might allow you to use existing capabilities and skillsets.

Avoid exposure to the internet with dedicated WAN links

Many organizations have chosen the hybrid IT route. With hybrid IT, some of the company's information assets are in Azure, and others remain on-premises. In many cases, some components of a service are running in Azure while other components remain on-premises.

In a hybrid IT scenario, there's usually some type of cross-premises connectivity. Cross-premises connectivity allows the company to connect its on-premises networks to Azure virtual networks. Two cross-premises connectivity solutions are available:

- [Site-to-site VPN](#). It's a trusted, reliable, and established technology, but the connection takes place over the internet. Bandwidth is constrained to a maximum of about 1.25 Gbps. Site-to-site VPN is a desirable option in some scenarios.
- [Azure ExpressRoute](#). We recommend that you use [ExpressRoute](#) for your cross-premises connectivity. ExpressRoute lets you extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services like Azure, Microsoft 365, and Dynamics 365. ExpressRoute is a dedicated WAN link between your on-premises location or a Microsoft Exchange hosting provider. Because this is a telco connection, your data doesn't travel over the internet, so it isn't exposed to the potential risks of internet communications.

The location of your ExpressRoute connection can affect firewall capacity, scalability, reliability, and network traffic visibility. You'll need to identify where to terminate ExpressRoute in existing (on-premises) networks. You can:

- Terminate outside the firewall (the perimeter network paradigm). Use this recommendation if you require visibility into the traffic, if you need to continue an existing practice of isolating datacenters, or if you're solely putting extranet resources on Azure.
- Terminate inside the firewall (the network extension paradigm). This is the default recommendation. In all other cases, we recommend treating Azure as another datacenter.

Optimize uptime and performance

If a service is down, information can't be accessed. If performance is so poor that the data is unusable, you can consider the data to be inaccessible. From a security

perspective, you need to do whatever you can to make sure that your services have optimal uptime and performance.

A popular and effective method for enhancing availability and performance is load balancing. Load balancing is a method of distributing network traffic across servers that are part of a service. For example, if you have front-end web servers as part of your service, you can use load balancing to distribute the traffic across your multiple front-end web servers.

This distribution of traffic increases availability because if one of the web servers becomes unavailable, the load balancer stops sending traffic to that server and redirects it to the servers that are still online. Load balancing also helps performance, because the processor, network, and memory overhead for serving requests is distributed across all the load-balanced servers.

We recommend that you employ load balancing whenever you can, and as appropriate for your services. Following are scenarios at both the Azure virtual network level and the global level, along with load-balancing options for each.

Scenario: You have an application that:

- Requires requests from the same user/client session to reach the same back-end virtual machine. Examples of this are shopping cart apps and web mail servers.
- Accepts only a secure connection, so unencrypted communication to the server isn't an acceptable option.
- Requires multiple HTTP requests on the same long-running TCP connection to be routed or load balanced to different back-end servers.

Load-balancing option: Use [Azure Application Gateway](#), an HTTP web traffic load balancer. Application Gateway supports end-to-end TLS encryption and [TLS termination](#) at the gateway. Web servers can then be unburdened from encryption and decryption overhead and traffic flowing unencrypted to the back-end servers.

Scenario: You need to load balance incoming connections from the internet among your servers located in an Azure virtual network. Scenarios are when you:

- Have stateless applications that accept incoming requests from the internet.
- Don't require sticky sessions or TLS offload. Sticky sessions is a method used with Application Load Balancing, to achieve server-affinity.

Load-balancing option: Use the Azure portal to [create an external load balancer](#) that spreads incoming requests across multiple VMs to provide a higher level of availability.

Scenario: You need to load balance connections from VMs that are not on the internet. In most cases, the connections that are accepted for load balancing are initiated by devices on an Azure virtual network, such as SQL Server instances or internal web servers.

Load-balancing option: Use the Azure portal to [create an internal load balancer](#) that spreads incoming requests across multiple VMs to provide a higher level of availability.

Scenario: You need global load balancing because you:

- Have a cloud solution that is widely distributed across multiple regions and requires the highest level of uptime (availability) possible.
- Need the highest level of uptime possible to make sure that your service is available even if an entire datacenter becomes unavailable.

Load-balancing option: Use Azure Traffic Manager. Traffic Manager makes it possible to load balance connections to your services based on the location of the user.

For example, if the user makes a request to your service from the EU, the connection is directed to your services located in an EU datacenter. This part of Traffic Manager global load balancing helps to improve performance because connecting to the nearest datacenter is faster than connecting to datacenters that are far away.

Disable RDP/SSH Access to virtual machines

It's possible to reach Azure virtual machines by using [Remote Desktop Protocol \(RDP\)](#) and the [Secure Shell](#) (SSH) protocol. These protocols enable the management VMs from remote locations and are standard in datacenter computing.

The potential security problem with using these protocols over the internet is that attackers can use [brute force](#) techniques to gain access to Azure virtual machines.

After the attackers gain access, they can use your VM as a launch point for compromising other machines on your virtual network or even attack networked devices outside Azure.

We recommend that you disable direct RDP and SSH access to your Azure virtual machines from the internet. After direct RDP and SSH access from the internet is disabled, you have other options that you can use to access these VMs for remote management.

Scenario: Enable a single user to connect to an Azure virtual network over the internet.

Option: [Point-to-site VPN](#) is another term for a remote access VPN client/server connection. After the point-to-site connection is established, the user can use RDP or SSH to connect to any VMs located on the Azure virtual network that the user

connected to via point-to-site VPN. This assumes that the user is authorized to reach those VMs.

Point-to-site VPN is more secure than direct RDP or SSH connections because the user has to authenticate twice before connecting to a VM. First, the user needs to authenticate (and be authorized) to establish the point-to-site VPN connection. Second, the user needs to authenticate (and be authorized) to establish the RDP or SSH session.

Scenario: Enable users on your on-premises network to connect to VMs on your Azure virtual network.

Option: A [site-to-site VPN](#) connects an entire network to another network over the internet. You can use a site-to-site VPN to connect your on-premises network to an Azure virtual network. Users on your on-premises network connect by using the RDP or SSH protocol over the site-to-site VPN connection. You don't have to allow direct RDP or SSH access over the internet.

Scenario: Use a dedicated WAN link to provide functionality similar to the site-to-site VPN.

Option: Use [ExpressRoute](#). It provides functionality similar to the site-to-site VPN. The main differences are:

- The dedicated WAN link doesn't traverse the internet.
- Dedicated WAN links are typically more stable and perform better.

Secure your critical Azure service resources to only your virtual networks

Use Azure Private Link to access Azure PaaS Services (for example, Azure Storage and SQL Database) over a private endpoint in your virtual network. Private Endpoints allow you to secure your critical Azure service resources to only your virtual networks. Traffic from your virtual network to the Azure service always remains on the Microsoft Azure backbone network. Exposing your virtual network to the public internet is no longer necessary to consume Azure PaaS Services.

Azure Private Link provides the following benefits:

- **Improved security for your Azure service resources:** With Azure Private Link, Azure service resources can be secured to your virtual network using private endpoint. Securing service resources to a private endpoint in virtual network provides improved security by fully removing public internet access to resources, and allowing traffic only from private endpoint in your virtual network.

- **Privately access Azure service resources on the Azure platform:** Connect your virtual network to services in Azure using private endpoints. There's no need for a public IP address. The Private Link platform will handle the connectivity between the consumer and services over the Azure backbone network.
- **Access from On-premises and peered networks:** Access services running in Azure from on-premises over ExpressRoute private peering, VPN tunnels, and peered virtual networks using private endpoints. There's no need to configure ExpressRoute Microsoft peering or traverse the internet to reach the service. Private Link provides a secure way to migrate workloads to Azure.
- **Protection against data leakage:** A private endpoint is mapped to an instance of a PaaS resource instead of the entire service. Consumers can only connect to the specific resource. Access to any other resource in the service is blocked. This mechanism provides protection against data leakage risks.
- **Global reach:** Connect privately to services running in other regions. The consumer's virtual network could be in region A and it can connect to services in region B.
- **Simple to set up and manage:** You no longer need reserved, public IP addresses in your virtual networks to secure Azure resources through an IP firewall. There are no NAT or gateway devices required to set up the private endpoints. Private endpoints are configured through a simple workflow. On service side, you can also manage the connection requests on your Azure service resource with ease. Azure Private Link works for consumers and services belonging to different Microsoft Entra tenants too.

To learn more about private endpoints and the Azure services and regions that private endpoints are available for, see [Azure Private Link](#).

Next steps

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | Get help at Microsoft Q&A

Azure Operational Security best practices

Article • 06/27/2024

This article provides a set of operational best practices for protecting your data, applications, and other assets in Azure.

The best practices are based on a consensus of opinion, and they work with current Azure platform capabilities and feature sets. Opinions and technologies change over time and this article is updated on a regular basis to reflect those changes.

Define and deploy strong operational security practices

Azure operational security refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Azure. Azure operational security is built on a framework that incorporates the knowledge gained through capabilities that are unique to Microsoft, including the [Security Development Lifecycle \(SDL\)](#), the [Microsoft Security Response Center](#) program, and deep awareness of the cybersecurity threat landscape.

Enforce multifactor verification for users

We recommend that you require two-step verification for all of your users. This includes administrators and others in your organization who can have a significant impact if their account is compromised (for example, financial officers).

There are multiple options for requiring two-step verification. The best option for you depends on your goals, the Microsoft Entra edition you're running, and your licensing program. See [How to require two-step verification for a user](#) to determine the best option for you. See the [Microsoft Entra ID](#) and [Microsoft Entra multifactor Authentication](#) pricing pages for more information about licenses and pricing.

Following are options and benefits for enabling two-step verification:

Option 1: Enable MFA for all users and login methods with Microsoft Entra Security Defaults **Benefit:** This option enables you to easily and quickly enforce MFA for all users in your environment with a stringent policy to:

- Challenge administrative accounts and administrative logon mechanisms

- Require MFA challenge via Microsoft Authenticator for all users
- Restrict legacy authentication protocols.

This method is available to all licensing tiers but is not able to be mixed with existing Conditional Access policies. You can find more information in [Microsoft Entra Security Defaults](#)

Option 2: Enable multifactor authentication by changing user state.

Benefit: This is the traditional method for requiring two-step verification. It works with both [Microsoft Entra multifactor authentication in the cloud](#) and [Azure Multi-Factor Authentication Server](#). Using this method requires users to perform two-step verification every time they sign in and overrides Conditional Access policies.

To determine where multifactor authentication needs to be enabled, see [Which version of Microsoft Entra multifactor authentication is right for my organization?](#).

Option 3: Enable multifactor authentication with Conditional Access policy. **Benefit:** This option allows you to prompt for two-step verification under specific conditions by using [Conditional Access](#). Specific conditions can be user sign-in from different locations, untrusted devices, or applications that you consider risky. Defining specific conditions where you require two-step verification enables you to avoid constant prompting for your users, which can be an unpleasant user experience.

This is the most flexible way to enable two-step verification for your users. Enabling a Conditional Access policy works only for Microsoft Entra multifactor authentication in the cloud and is a premium feature of Microsoft Entra ID. You can find more information on this method in [Deploy cloud-based Microsoft Entra multifactor authentication](#).

Option 4: Enable multifactor authentication with Conditional Access policies by evaluating Risk-based Conditional Access policies.

Benefit: This option enables you to:

- Detect potential vulnerabilities that affect your organization's identities.
- Configure automated responses to detected suspicious actions that are related to your organization's identities.
- Investigate suspicious incidents and take appropriate action to resolve them.

This method uses the Microsoft Entra ID Protection risk evaluation to determine if two-step verification is required based on user and sign-in risk for all cloud applications. This method requires Microsoft Entra ID P2 licensing. You can find more information on this method in [Microsoft Entra ID Protection](#).

 **Note**

Option 2, enabling multifactor authentication by changing the user state, overrides Conditional Access policies. Because options 3 and 4 use Conditional Access policies, you cannot use option 2 with them.

Organizations that don't add extra layers of identity protection, such as two-step verification, are more susceptible for credential theft attack. A credential theft attack can lead to data compromise.

Manage and monitor user passwords

The following table lists some best practices related to managing user passwords:

Best practice: Ensure you have the proper level of password protection in the cloud.

Detail: Follow the guidance in [Microsoft Password Guidance](#), which is scoped to users of the Microsoft identity platforms (Microsoft Entra ID, Active Directory, and Microsoft account).

Best practice: Monitor for suspicious actions related to your user accounts.

Detail: Monitor for [users at risk](#) and [risky sign-ins](#) by using Microsoft Entra security reports.

Best practice: Automatically detect and remediate high-risk passwords.

Detail: [Microsoft Entra ID Protection](#) is a feature of the Microsoft Entra ID P2 edition that enables you to:

- Detect potential vulnerabilities that affect your organization's identities
- Configure automated responses to detected suspicious actions that are related to your organization's identities
- Investigate suspicious incidents and take appropriate actions to resolve them

Receive incident notifications from Microsoft

Be sure your security operations team receives Azure incident notifications from Microsoft. An incident notification lets your security team know you have compromised Azure resources so they can quickly respond to and remediate potential security risks.

In the Azure enrollment portal, you can ensure admin contact information includes details that notify security operations. Contact information is an email address and phone number.

Organize Azure subscriptions into management groups

If your organization has many subscriptions, you might need a way to efficiently manage access, policies, and compliance for those subscriptions. [Azure management groups](#) provide a level of scope that's above subscriptions. You organize subscriptions into containers called management groups and apply your governance conditions to the management groups. All subscriptions within a management group automatically inherit the conditions applied to the management group.

You can build a flexible structure of management groups and subscriptions into a directory. Each directory is given a single top-level management group called the root management group. This root management group is built into the hierarchy to have all management groups and subscriptions fold up to it. The root management group allows global policies and Azure role assignments to be applied at the directory level.

Here are some best practices for using management groups:

Best practice: Ensure that new subscriptions apply governance elements like policies and permissions as they are added.

Detail: Use the root management group to assign enterprise-wide security elements that apply to all Azure assets. Policies and permissions are examples of elements.

Best practice: Align the top levels of management groups with segmentation strategy to provide a point for control and policy consistency within each segment.

Detail: Create a single management group for each segment under the root management group. Don't create any other management groups under the root.

Best practice: Limit management group depth to avoid confusion that hampers both operations and security.

Detail: Limit your hierarchy to three levels, including the root.

Best practice: Carefully select which items to apply to the entire enterprise with the root management group.

Detail: Ensure root management group elements have a clear need to be applied across every resource and that they're low impact.

Good candidates include:

- Regulatory requirements that have a clear business impact (for example, restrictions related to data sovereignty)
- Requirements with near-zero potential negative effect on operations, like policy with audit effect or Azure RBAC permission assignments that have been carefully

reviewed

Best practice: Carefully plan and test all enterprise-wide changes on the root management group before applying them (policy, Azure RBAC model, and so on).

Detail: Changes in the root management group can affect every resource on Azure. While they provide a powerful way to ensure consistency across the enterprise, errors or incorrect usage can negatively affect production operations. Test all changes to the root management group in a test lab or production pilot.

Streamline environment creation with blueprints

The [Azure Blueprints](#) service enables cloud architects and central information technology groups to define a repeatable set of Azure resources that implements and adheres to an organization's standards, patterns, and requirements. Azure Blueprints makes it possible for development teams to rapidly build and stand up new environments with a set of built-in components and the confidence that they're creating those environments within organizational compliance.

Monitor storage services for unexpected changes in behavior

Diagnosing and troubleshooting issues in a distributed application hosted in a cloud environment can be more complex than it is in traditional environments. Applications can be deployed in a PaaS or IaaS infrastructure, on-premises, on a mobile device, or in some combination of these environments. Your application's network traffic might traverse public and private networks, and your application might use multiple storage technologies.

You should continuously monitor the storage services that your application uses for any unexpected changes in behavior (such as slower response times). Use logging to collect more detailed data and to analyze a problem in depth. The diagnostics information that you obtain from both monitoring and logging helps you to determine the root cause of the issue that your application encountered. Then you can troubleshoot the issue and determine the appropriate steps to remediate it.

[Azure Storage Analytics](#) performs logging and provides metrics data for an Azure storage account. We recommend that you use this data to trace requests, analyze usage trends, and diagnose issues with your storage account.

Prevent, detect, and respond to threats

[Microsoft Defender for Cloud](#) helps you prevent, detect, and respond to threats by providing increased visibility into (and control over) the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with various security solutions.

The Free tier of Defender for Cloud offers limited security for your resources in Azure as well as Arc-enabled resources outside of Azure. The Enhanced Security Features extend these capabilities to include threat and vulnerability management, as well as regulatory compliance reporting. Defender for Cloud Plans help you find and fix security vulnerabilities, apply access and application controls to block malicious activity, detect threats by using analytics and intelligence, and respond quickly when under attack. You can try Defender for Cloud Standard at no cost for the first 30 days. We recommend that you [enable enhanced security features on your Azure subscriptions in Defender for Cloud](#).

Use Defender for Cloud to get a central view of the security state of all your resources in your own data centers, Azure and other clouds. At a glance, verify that the appropriate security controls are in place and configured correctly, and quickly identify any resources that need attention.

Defender for Cloud also integrates with [Microsoft Defender for Endpoint](#), which provides comprehensive Endpoint Detection and Response (EDR) capabilities. With Microsoft Defender for Endpoint integration, you can spot abnormalities and detect vulnerabilities. You can also detect and respond to advanced attacks on server endpoints monitored by Defender for Cloud.

Almost all enterprise organizations have a security information and event management (SIEM) system to help identify emerging threats by consolidating log information from diverse signal gathering devices. The logs are then analyzed by a data analytics system to help identify what's "interesting" from the noise that is inevitable in all log gathering and analytics solutions.

[Microsoft Sentinel](#) is a scalable, cloud-native, security information and event management (SIEM) and security orchestration automated response (SOAR) solution. Microsoft Sentinel provides intelligent security analytics and threat intelligence via alert detection, threat visibility, proactive hunting, and automated threat response.

Here are some best practices for preventing, detecting, and responding to threats:

Best practice: Increase the speed and scalability of your SIEM solution by using a cloud-based SIEM.

Detail: Investigate the features and capabilities of [Microsoft Sentinel](#) and compare them with the capabilities of what you're currently using on-premises. Consider adopting Microsoft Sentinel if it meets your organization's SIEM requirements.

Best practice: Find the most serious security vulnerabilities so you can prioritize investigation.

Detail: Review your [Azure secure score](#) to see the recommendations resulting from the Azure policies and initiatives built into Microsoft Defender for Cloud. These recommendations help address top risks like security updates, endpoint protection, encryption, security configurations, missing WAF, internet-connected VMs, and many more.

The secure score, which is based on Center for Internet Security (CIS) controls, lets you benchmark your organization's Azure security against external sources. External validation helps validate and enrich your team's security strategy.

Best practice: Monitor the security posture of machines, networks, storage and data services, and applications to discover and prioritize potential security issues.

Detail: Follow the [security recommendations](#) in Defender for Cloud starting with the highest priority items.

Best practice: Integrate Defender for Cloud alerts into your security information and event management (SIEM) solution.

Detail: Most organizations with a SIEM use it as a central clearinghouse for security alerts that require an analyst response. Processed events produced by Defender for Cloud are published to the Azure Activity Log, one of the logs available through Azure Monitor. Azure Monitor offers a consolidated pipeline for routing any of your monitoring data into a SIEM tool. See [Stream alerts to a SIEM, SOAR, or IT Service Management solution](#) for instructions. If you're using Microsoft Sentinel, see [Connect Microsoft Defender for Cloud](#).

Best practice: Integrate Azure logs with your SIEM.

Detail: Use [Azure Monitor to gather and export data](#). This practice is critical for enabling security incident investigation, and online log retention is limited. If you're using Microsoft Sentinel, see [Connect data sources](#).

Best practice: Speed up your investigation and hunting processes and reduce false positives by integrating Endpoint Detection and Response (EDR) capabilities into your attack investigation.

Detail: [Enable the Microsoft Defender for Endpoint integration](#) via your Defender for

Cloud security policy. Consider using Microsoft Sentinel for threat hunting and incident response.

Monitor end-to-end scenario-based network monitoring

Customers build an end-to-end network in Azure by combining network resources like a virtual network, ExpressRoute, Application Gateway, and load balancers. Monitoring is available on each of the network resources.

[Azure Network Watcher](#) is a regional service. Use its diagnostic and visualization tools to monitor and diagnose conditions at a network scenario level in, to, and from Azure.

The following are best practices for network monitoring and available tools.

Best practice: Automate remote network monitoring with packet capture.

Detail: Monitor and diagnose networking issues without logging in to your VMs by using Network Watcher. Trigger [packet capture](#) by setting alerts and gain access to real-time performance information at the packet level. When you see an issue, you can investigate in detail for better diagnoses.

Best practice: Gain insight into your network traffic by using flow logs.

Detail: Build a deeper understanding of your network traffic patterns by using [network security group flow logs](#). Information in flow logs helps you gather data for compliance, auditing, and monitoring your network security profile.

Best practice: Diagnose VPN connectivity issues.

Detail: Use Network Watcher to [diagnose your most common VPN Gateway and connection issues](#). You can not only identify the issue but also use detailed logs to further investigate.

Secure deployment by using proven DevOps tools

Use the following DevOps best practices to ensure that your enterprise and teams are productive and efficient.

Best practice: Automate the build and deployment of services.

Detail: [Infrastructure as code](#) is a set of techniques and practices that help IT pros remove the burden of day-to-day build and management of modular infrastructure. It

enables IT pros to build and maintain their modern server environment in a way that's like how software developers build and maintain application code.

You can use [Azure Resource Manager](#) to provision your applications by using a declarative template. In a single template, you can deploy multiple services along with their dependencies. You use the same template to repeatedly deploy your application in every stage of the application lifecycle.

Best practice: Automatically build and deploy to Azure web apps or cloud services.

Detail: You can configure your Azure DevOps Projects to [automatically build and deploy](#) to Azure web apps or cloud services. Azure DevOps automatically deploys the binaries after doing a build to Azure after every code check-in. The package build process is equivalent to the Package command in Visual Studio, and the publishing steps are equivalent to the Publish command in Visual Studio.

Best practice: Automate release management.

Detail: [Azure Pipelines](#) is a solution for automating multiple-stage deployment and managing the release process. Create managed continuous deployment pipelines to release quickly, easily, and often. With Azure Pipelines, you can automate your release process, and you can have predefined approval workflows. Deploy on-premises and to the cloud, extend, and customize as required.

Best practice: Check your app's performance before you launch it or deploy updates to production.

Detail: Run cloud-based [load tests](#) to:

- Find performance problems in your app.
- Improve deployment quality.
- Make sure that your app is always available.
- Make sure that your app can handle traffic for your next launch or marketing campaign.

[Apache JMeter](#) is a free, popular open source tool with a strong community backing.

Best practice: Monitor application performance.

Detail: [Azure Application Insights](#) is an extensible application performance management (APM) service for web developers on multiple platforms. Use Application Insights to monitor your live web application. It automatically detects performance anomalies. It includes analytics tools to help you diagnose issues and to understand what users actually do with your app. It's designed to help you continuously improve performance and usability.

Mitigate and protect against DDoS

Distributed denial of service (DDoS) is a type of attack that tries to exhaust application resources. The goal is to affect the application's availability and its ability to handle legitimate requests. These attacks are becoming more sophisticated and larger in size and impact. They can be targeted at any endpoint that is publicly reachable through the internet.

Designing and building for DDoS resiliency requires planning and designing for a variety of failure modes. Following are best practices for building DDoS-resilient services on Azure.

Best practice: Ensure that security is a priority throughout the entire lifecycle of an application, from design and implementation to deployment and operations. Applications can have bugs that allow a relatively low volume of requests to use a lot of resources, resulting in a service outage.

Detail: To help protect a service running on Microsoft Azure, you should have a good understanding of your application architecture and focus on the [five pillars of software quality](#). You should know typical traffic volumes, the connectivity model between the application and other applications, and the service endpoints that are exposed to the public internet.

Ensuring that an application is resilient enough to handle a denial of service that's targeted at the application itself is most important. Security and privacy are built into the Azure platform, beginning with the [Security Development Lifecycle \(SDL\)](#). The SDL addresses security at every development phase and ensures that Azure is continually updated to make it even more secure.

Best practice: Design your applications to [scale horizontally](#) to meet the demand of an amplified load, specifically in the event of a DDoS attack. If your application depends on a single instance of a service, it creates a single point of failure. Provisioning multiple instances makes your system more resilient and more scalable.

Detail: For [Azure App Service](#), select an [App Service plan](#) that offers multiple instances.

For Azure Cloud Services, configure each of your roles to use [multiple instances](#).

For [Azure Virtual Machines](#), ensure that your VM architecture includes more than one VM and that each VM is included in an [availability set](#). We recommend using Virtual Machine Scale Sets for autoscaling capabilities.

Best practice: Layering security defenses in an application reduces the chance of a successful attack. Implement secure designs for your applications by using the built-in capabilities of the Azure platform.

Detail: The risk of attack increases with the size (surface area) of the application. You can reduce the surface area by using an approval list to close down the exposed IP address space and listening ports that are not needed on the load balancers ([Azure Load Balancer](#) and [Azure Application Gateway](#)).

[Network security groups](#) are another way to reduce the attack surface. You can use service tags and [application security groups](#) to minimize complexity for creating security rules and configuring network security, as a natural extension of an application's structure.

You should deploy Azure services in a [virtual network](#) whenever possible. This practice allows service resources to communicate through private IP addresses. Azure service traffic from a virtual network uses public IP addresses as source IP addresses by default.

Using [service endpoints](#) switches service traffic to use virtual network private addresses as the source IP addresses when they're accessing the Azure service from a virtual network.

We often see customers' on-premises resources getting attacked along with their resources in Azure. If you're connecting an on-premises environment to Azure, minimize exposure of on-premises resources to the public internet.

Azure has two DDoS [service offerings](#) that provide protection from network attacks:

- Basic protection is integrated into Azure by default at no additional cost. The scale and capacity of the globally deployed Azure network provides defense against common network-layer attacks through always-on traffic monitoring and real-time mitigation. Basic requires no user configuration or application changes and helps protect all Azure services, including PaaS services like Azure DNS.
- Standard protection provides advanced DDoS mitigation capabilities against network attacks. It's automatically tuned to protect your specific Azure resources. Protection is simple to enable during the creation of virtual networks. It can also be done after creation and requires no application or resource changes.

Enable Azure Policy

[Azure Policy](#) is a service in Azure that you use to create, assign, and manage policies. These policies enforce rules and effects over your resources, so those resources stay compliant with your corporate standards and service-level agreements. Azure Policy meets this need by evaluating your resources for non-compliance with assigned policies.

Enable Azure Policy to monitor and enforce your organization's written policy. This will ensure compliance with your company or regulatory security requirements by centrally

managing security policies across your hybrid cloud workloads. Learn how to [create and manage policies to enforce compliance](#). See [Azure Policy definition structure](#) for an overview of the elements of a policy.

Here are some security best practices to follow after you adopt Azure Policy:

Best practice: Policy supports several types of effects. You can read about them in [Azure Policy definition structure](#). Business operations can be negatively affected by the **deny** effect and the **remediate** effect, so start with the **audit** effect to limit the risk of negative impact from policy.

Detail: [Start policy deployments in audit mode](#) and then later progress to **deny** or **remediate**. Test and review the results of the audit effect before you move to **deny** or **remediate**.

For more information, see [Create and manage policies to enforce compliance](#).

Best practice: Identify the roles responsible for monitoring for policy violations and ensuring the right remediation action is taken quickly.

Detail: Have the assigned role monitor compliance through the [Azure portal](#) or via the [command line](#).

Best practice: Azure Policy is a technical representation of an organization's written policies. Map all Azure Policy definitions to organizational policies to reduce confusion and increase consistency.

Detail: Document mapping in your organization's documentation or in the Azure Policy definition itself by adding a reference to the organizational policy in the [policy definition](#) or the [initiative definition](#) description.

Monitor Microsoft Entra risk reports

The vast majority of security breaches take place when attackers gain access to an environment by stealing a user's identity. Discovering compromised identities is no easy task. Microsoft Entra ID uses adaptive machine learning algorithms and heuristics to detect suspicious actions that are related to your user accounts. Each detected suspicious action is stored in a record called a [risk detection](#). Risk detections are recorded in Microsoft Entra security reports. For more information, read about the [users at risk security report](#) and the [risky sign-ins security report](#).

Next steps

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

The following resources are available to provide more general information about Azure security and related Microsoft services:

- [Azure Security Team Blog](#) - for up to date information on the latest in Azure Security
 - [Microsoft Security Response Center](#) - where Microsoft security vulnerabilities, including issues with Azure, can be reported or via email to secure@microsoft.com
-

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#)

Azure landing zone frequently asked questions (FAQ)

Article • 12/12/2023

This article answers frequently asked questions about Azure landing zone architecture.

For FAQs about **implementing Azure landing zone architecture**, see [Enterprise-scale implementation FAQ ↗](#).

What is the Azure landing zone accelerator?

The Azure landing zone accelerator is an Azure portal-based deployment experience. It deploys an opinionated implementation based on the [Azure landing zone conceptual architecture](#).

Which are the recommended accelerators and implementations for Azure landing zones?

Microsoft actively develops and maintains the platform and application accelerators and implementations in alignment with the Azure landing zone [design principles](#) and [design area](#) guidance.

Review the [Deploy Azure landing zones](#) guidance to learn more about the recommended platform and application landing zones.

To learn how to tailor your Azure landing zones deployment to meet your needs, see [Tailor the Azure landing zone architecture to meet requirements](#)

💡 Tip

To request an addition to the accelerator and implementation list, raise a GitHub issue on the [ALZ repository ↗](#).

What is the Azure landing zone conceptual architecture?

The Azure landing zone conceptual architecture represents scale and maturity decisions. It's based on lessons learned and feedback from customers who have adopted Azure as part of their digital estate. This conceptual architecture can help your organization set a direction for designing and implementing a landing zone.

What does a landing zone map to in Azure in the context of Azure landing zone architecture?

From an Azure landing zone point of view, landing zones are individual Azure subscriptions.

What does policy-driven governance mean, and how does it work?

[Policy-driven governance](#) is one of the key design principles of enterprise-scale architecture.

Policy-driven governance means using Azure Policy to reduce the time you need for common and repeated operational tasks across your Azure tenant. Use many of the [Azure Policy effects](#), such as `Append`, `Deny`, `DeployIfNotExists`, and `Modify`, to prevent non-compliance by either restricting non-compliant resources (as defined by the policy definition) from being created or updated or by deploying resources or modifying settings of a resource creation or update request to make them compliant. Some effects, such as `Audit`, `Disabled`, and `AuditIfNotExists`, don't prevent or take action; they only audit and report on non-compliance.

Some examples of policy-driven governance are:

- `Deny` effect: Prevents subnets from being created or updated to have no Network Security Groups associated with them.
- `DeployIfNotExists` effect: A new subscription (landing zone) is created and placed into a management group within your Azure landing zone deployment. Azure Policy ensures that Microsoft Defender for Cloud (formerly known as Azure Security Center) is enabled on the subscription. It also configures the diagnostic settings for Activity Log to send logs to the Log Analytics workspace in the Management subscription.

Instead of repeating code or manual activities when a new subscription is created, the `DeployIfNotExists` policy definition automatically deploys and configures them

for you.

What if we can't or aren't yet ready to utilize DeployIfNotExists (DINE) policies?

We have a dedicated page that walks through the various phases and options you have to either "disable" DINE policies or use our three phase approach to adopt them over time within your environment.

See the guidance [Adopting policy driven guardrails](#)

Should we use Azure Policy to deploy workloads?

In short, **no**. Use Azure Policy to control, govern, and keep your workloads and landing zones compliant. It isn't designed to deploy entire workloads and other tooling. Use the Azure portal or infrastructure-as-code offerings (ARM Templates, Bicep, Terraform) to deploy and manage your workload and get the autonomy you need.

What is Cloud Adoption Framework Landing zones for Terraform (aztfmod)?

The Cloud Adoption Framework landing zones [open source project \(OSS\)](#) (also known as *aztfmod*) is a community driven project owned and maintained outside of the Azure landing zone core team and the Azure GitHub organization. If your organization chooses to use this OSS project, consideration should be given to the support available as this is driven by the community effort through GitHub.

What if we already have resources in our landing zones and later assign an Azure Policy definition that includes them in its scope?

Review the following documentation sections:

- [Transition existing Azure environments to the Azure landing zone conceptual architecture - "Policy" section](#)
- [Quickstart: Create a policy assignment to identify non-compliant resources - "Identify non-compliant resources" section](#)

How do we handle "dev/test/production" workload landing zones in Azure landing zone architecture?

For more information, see [Manage application development environments in Azure landing zones](#).

Why are we asked to specify Azure regions during the Azure landing zone accelerator deployment and what are they used for?

When you deploy Azure landing zone architecture by using the Azure landing zone accelerator portal-based experience, select an Azure region to deploy into. The first tab, **Deployment location**, determines where the deployment data is stored. For more information, see [Tenant deployments with ARM templates](#). Some parts of a landing zone are deployed globally but their deployment metadata is tracked in a regional metadata store. The metadata regarding their deployment is stored in the region selected on the **Deployment location** tab.

The region selector on the **Deployment location** tab is also used to select which Azure region any region-specific resources should be stored, such as a Log Analytics workspace and an automation account, if required.

If you deploy a networking topology on the **Network topology and connectivity** tab, you need to select an Azure region to deploy the networking resources to. This region can be different from the region selected on the **Deployment location** tab.

For more information about the regions that landing zone resources use, see [Landing zone regions](#).

How do we enable more Azure regions when we use Azure landing zone architecture?

To understand how to add new regions to a landing zone, or how to move landing zone resources to a different region, see [Landing zone regions](#).

Should we create a new Azure Subscription every time or should we reuse Azure Subscriptions?

What is subscription reuse?

Subscription reuse is the process of reissuing an existing subscription to a new owner. There should be a process to reset the subscription to a known clean state and then reassigned to a new owner.

Why should I consider reusing subscriptions?

In general, we recommend that customers adopt the [Subscription Democratization design principle](#). However, there are specific circumstances where subscription reuse isn't possible or recommended.

Tip

Watch the YouTube video on the Subscription Democratization design principle here: [Azure Landing Zones - How many subscriptions should I use in Azure?](#) ↗

You should consider subscription reuse if you meet one of the following circumstances:

- You have an Enterprise Agreement (EA) and plan to create more than 5,000 subscriptions on a single EA Account Owner Account (billing account), including deleted subscriptions.
- You have a Microsoft Customer Agreement (MCA) or Microsoft Partner Agreement MPA and plan to have more than 5,000 active subscriptions
- You're a pay-as-you-go customer
- You use a Microsoft Azure Sponsorship
- You commonly create:
 1. Ephemeral lab or sandbox environments
 2. Demo environments for proofs-of-concept (POCs) or minimum viable products (MVP), including independent software vendors (ISV) for customer demo/trial access
 3. Training environments, such as MSPs/Trainer's learner environments

How do I reuse subscriptions?

If you match one of the above scenarios or considerations, then you might need to consider reusing existing decommissioned or unused subscriptions and reassigning them to a new owner and purpose.

Clean up old subscription

You first need to clean up the old subscription for reuse. You need to perform the following actions on a subscription before it's ready for reuse:

- Remove Resource Groups and contained resources.
- Remove Role Assignments, including Privileged Identity Management (PIM) Role Assignments, at the subscription scope.
- Remove Custom Role-based Access Control (RBAC) Definitions, at the subscription scope.
- Remove Policy Definitions, Initiatives, Assignments and Exemptions at the subscription scope.
- Remove deployments at the subscription scope.
- Remove tags at the subscription scope.
- Remove any Resource Locks at the subscription scope.
- Remove any Microsoft Cost Management budgets at the subscription scope.
- Reset Microsoft Defender for Cloud plans to Free Tiers unless organizational requirements mandate these logs are set to the paid tiers. You normally enforce these requirements via Azure Policy.
- Remove subscription activity logs (diagnostic settings) forwarding to Log Analytics Workspaces, Event Hubs, Storage Account or other supported destinations unless organizational requirements mandate forwarding these logs while a subscription is active.
- Remove any Azure Lighthouse Delegations at the subscription scope.
- Remove any hidden resources from the subscription.

Tip

Using `Get-AzResource` or `az resource list -o table` targeted at the subscription scope will help you find any hidden or remaining resources to remove before re-assigning.

Reassign the subscription

You can reassign the subscription after you clean up the subscription. Here are some common activities that you might want to perform as part of the reassignment process:

- Add new tags and set values for them on the subscription.
- Add new Role Assignments, or Privileged Identity Management (PIM) Role Assignments, at the subscription scope for the new owners. Typically these assignments would be to Microsoft Entra groups instead of individuals.
- Place the subscription into the desired Management Group based on its governance requirements.
- Create new Microsoft Cost Management budgets and set alerts to new owners when thresholds met.
- Set Microsoft Defender for Cloud plans to desired Tiers. You should enforce this setting via Azure Policy once placed into the correct Management Group.
- Configure subscription activity logs (diagnostic settings) forwarding to Log Analytics Workspaces, Event Hubs, Storage Account or other supported destinations. You should enforce this setting via Azure Policy once placed into the correct Management Group.

What is a sovereign landing zone and how is it related to the Azure landing zone architecture?

The sovereign landing zone is a component of Microsoft Cloud for Sovereignty that's intended for public sector customers who need advanced sovereignty controls. As a tailored version of the Azure landing zone conceptual architecture, the sovereign landing zone aligns Azure capabilities such as service residency, customer-managed keys, Azure Private Link, and confidential computing. Through this alignment, the sovereign landing zone creates a cloud architecture where data and workloads offer encryption and protection from threats by default.

Note

Microsoft Cloud for Sovereignty is oriented toward government organizations with sovereignty needs. You should carefully consider whether you need the Microsoft Cloud for Sovereignty capabilities, and only then consider adopting the sovereign landing zone architecture.

For more information about the sovereign landing zone, see [Sovereignty considerations for Azure landing zones](#).

Skills readiness path during the readiness phase of a migration journey

Article • 12/01/2022

During the readiness phase of a migration journey, the objective is to prepare for the journey ahead. This phase is accomplished in two primary areas: organizational readiness and environmental (technical) readiness. Each area might require new skills for both technical and nontechnical contributors. The following sections describe a few options to help build the necessary skills.

Organizational readiness learning paths

Depending on the motivations and business outcomes associated with a cloud adoption effort, leaders might be required to establish new organizational structures or virtual teams to facilitate various functions. The following articles help to develop the skills that are necessary to structure those teams in accordance with desired outcomes:

- [Initial organization alignment](#): Overview of organizational alignment and various team structures to facilitate specific goals.
- [Break down silos and fiefdoms](#): Understand two common organizational antipatterns and ways to guide the team to productive collaboration.

Environmental (technical) readiness learning paths

During the readiness phase, technical staff are called upon to create a migration landing zone that's capable of hosting, operating, and governing workloads that were migrated to the cloud. Developing the necessary skills can be accelerated with the following learning paths:

- [Create an Azure account](#): The first step to using Azure is to create an account. Your account holds the Azure services you provision and handles your personal settings like identity, billing, and preferences.
- [Azure portal](#): Tour the Azure portal features and services, and customize the portal.
- [Introduction to Azure](#): Get started with Azure by creating and configuring your first virtual machine in the cloud.
- [Introduction to security in Azure](#): Discuss the basic concepts for protecting your infrastructure and data when you work in the cloud. Understand what

responsibilities are yours and what Azure takes care of for you.

- [Manage resources in Azure](#): Learn how to work with the Azure command line and web portal to create, manage, and control cloud-based resources.
- [Create a VM](#): Create a virtual machine by using the Azure portal.
- [Azure networking](#): Learn some of the Azure networking basics and how Azure networking helps improve resiliency and reduce latency.
- [Azure compute options](#): Review the Azure compute services.
- [Secure resources with Azure role-based access control \(Azure RBAC\)](#): Use Azure RBAC to secure resources.
- [Data storage options](#): Benefits of Azure data storage.

During the readiness phase, architects are called upon to architect solutions that span all Azure environments. The following skill-building resources can prepare architects for these tasks:

- [Foundations for Cloud Architecture](#): A Pluralsight course to help architect the right foundational solutions.
- [Microsoft Azure Architecture](#): A Pluralsight course to ground architects in Azure architecture.
- [Designing Migrations for Microsoft Azure](#): A Pluralsight course to help architects design a migration solution.

Deeper skills exploration

Various learning options beyond these initial options are available for developing skills.

Typical mappings of cloud IT roles

Microsoft and partners offer various options for all audiences to develop skills with Azure services.

- [Map roles and skills](#): A resource for mapping your cloud career path. Learn about your cloud role and suggested skills. Follow a learning curriculum at your own pace to build the skills that you need most to stay relevant.
- Explore [Azure certification training and exams](#) to gain official recognition for your Azure knowledge.

Microsoft Learn

Microsoft Learn is a new approach to learning. Readiness for the new skills and responsibilities that come with cloud adoption doesn't come easily. Microsoft Learn provides a more rewarding approach to hands-on learning that helps you achieve your goals faster. Earn points and levels and achieve more.

The following examples are a few tailored learning paths that align to the Ready methodology of the Cloud Adoption Framework:

[Azure fundamentals](#): Learn cloud concepts such as high availability, scalability, elasticity, agility, fault tolerance, and disaster recovery. Understand the benefits of cloud computing in Azure and how it can save you time and money. Compare and contrast basic strategies for transitioning to the Azure cloud. Explore the breadth of services available in Azure including compute, network, storage, and security.

[Manage resources in Azure](#): Learn how to work with the Azure command line and web portal to create, manage, and control cloud-based resources.

[Administer infrastructure resources in Azure](#): Learn how to create, manage, secure, and scale virtual machine resources.

[Store data in Azure](#): Azure provides a variety of ways to store data: unstructured, archival, relational, and more. Learn the basics of storage management in Azure, how to create a storage account, and how to choose the right model for the data you want to store in the cloud.

[Architect great solutions in Azure](#): Learn how to design and build secure, scalable, and high-performing solutions in Azure by examining the core principles found in sound architecture.

Learn more

For additional learning paths, browse the [Microsoft Learn training catalog](#). Use the **Roles** filter to align learning paths with your role.

Cloud readiness antipatterns

Article • 03/22/2023

Customers often experience antipatterns during the readiness phase of cloud adoption. These antipatterns can lead to unexpected downtime, disaster recovery problems, and availability issues.

Antipattern: Assume released services are ready for production

Because cloud computing is evolving rapidly, companies often release preview versions of new services. Customers tend to assume that they can use any available cloud service in a production environment. But, problems can result, for these reasons:

- Preview services usually don't provide uptime service-level agreements (SLAs).
- New services often aren't as mature as cloud services that are already available.

Example: Use a preview service in production

A research institute uses a preview cloud service in production. The service seems to be a good fit for its use case. But, the institute doesn't perform due diligence on the service. The institute also doesn't follow its reference architecture's requirements and guidelines.

Problems come up with the preview service that lead to unexpected downtime. The institute begins to think that cloud services in general aren't as mature or resilient as promised.

Preferred outcome: Use pre-approved cloud services in production

When evaluating new services that are in preview, only use these services in proof of concept (POC) scenarios. Don't use these services in production environments, because they don't have SLAs. Find the right balance between functionality and maturity when approving cloud services. See [Cloud services due diligence checklist](#) for an established framework that you can use to quickly evaluate cloud services.

Antipattern: Assume increased resiliency and availability

Cloud computing often offers advantages over on-premises computing. Examples include:

- **Increased resiliency:** Recovering after failure.
- **Availability:** Running in a healthy state without significant downtime.

Because most cloud services offer these advantages, many companies assume that all cloud services offer resiliency and high availability by default. In reality, these features are often only available at extra cost and with additional technical effort.

Example: Assume high availability

A start-up implements a mission-critical application on infrastructure as a service (IaaS) services. Developers at the start-up have looked into a virtual machine (VM) with an uptime SLA of 99.9%. Since they'd like to cut costs, they use a single VM and premium storage.

When the VM fails, their application can't recover. Unexpected downtime results. They'd assumed that the cloud offers high availability by default. They weren't aware that performance guarantees can differ between:

- Service models like platform as a service (PaaS) and software as a service (SaaS).
- Technical architectures like load-balanced availability sets and Availability Zones.

Preferred outcome: Reduce failures while balancing resiliency and costs

See trusted, mature resources for information on architectural best practices that can reduce the scope of failures:

- Reference architectures
- [Microsoft Azure Well-Architected Framework](#)

Identify the right balance between costs and features like [high resiliency and availability](#). Increased resiliency and availability typically lead to increased costs. For instance:

- A single VM might have an SLA with a guaranteed uptime of 99.9%.
- Two VMs running the same workload would provide an SLA with an uptime between 99.95 and 99.99 percent.

Engage in the essential process of *requirements engineering* when designing a cloud-based solution. Use an [SLA estimator](#) to help calculate your application's end-to-end SLA.

Antipattern: Become a cloud provider

Some companies try to make their internal IT department a cloud provider. IT then becomes responsible for reference architectures. IT also needs to provide IaaS and PaaS to business units. Since this type of work isn't usually part of IT's core business, the resulting service offerings can lack usability, resiliency, efficiency, and security.

Example: Provide monolithic managed cloud services

A corporation's IT department establishes a cloud center of excellence (CCoE) that serves as a broker between IT and business units. To ensure the corporation is cloud-compliant, the managing board assigns the CCoE the task of providing monolithic end-to-end services. The CCoE sets up an internal cloud procurement portal that business units can use to order a fully managed cloud VM as a service. But, IT controls who can access and use the entire platform. As a result, IT actively prevents business units from taking advantage of the full range of services that Azure provides. Business units can't access the cloud portal. They only get access through Secure Shell (SSH) and Remote Desktop Protocol (RDP) to the server that they order.

For several reasons, the CCoE then has trouble providing a monolithic managed service to wrap each service that's available in the cloud:

- The cloud offers a large number of services across multiple solution areas. Compared with developing IaaS solutions, designing and engineering Internet of Things (IoT) and AI solutions requires different expertise and skill sets.
- Cloud services change frequently.
- Trying to provide monolithic services increases the time to market substantially, with IT managing the process, not the business units.

Preferred outcome: Provide guardrails

When adopting cloud technologies, have the IT department gain firsthand experience with the cloud by starting with IT workloads. Use the [Microsoft Cloud Adoption Framework for Azure](#) to identify your [first adoption project](#).

Use a mature [cloud operating model](#) such as [centralized operations](#) that makes IT responsible for defining platform guardrails like governance. Then business units can

adopt cloud projects in a secure and consistent manner, within the guardrails that IT defines.

Consider adopting only one major public cloud provider at the start, because all major platforms differ significantly in setup, management, and usage.

Use SaaS solutions as much as possible for IT tooling, such as:

- Code repositories.
- Continuous integration and continuous delivery (CI/CD).
- Collaboration systems.

For cloud workloads, advise IT to use familiar procedures that operate safely and securely at scale.

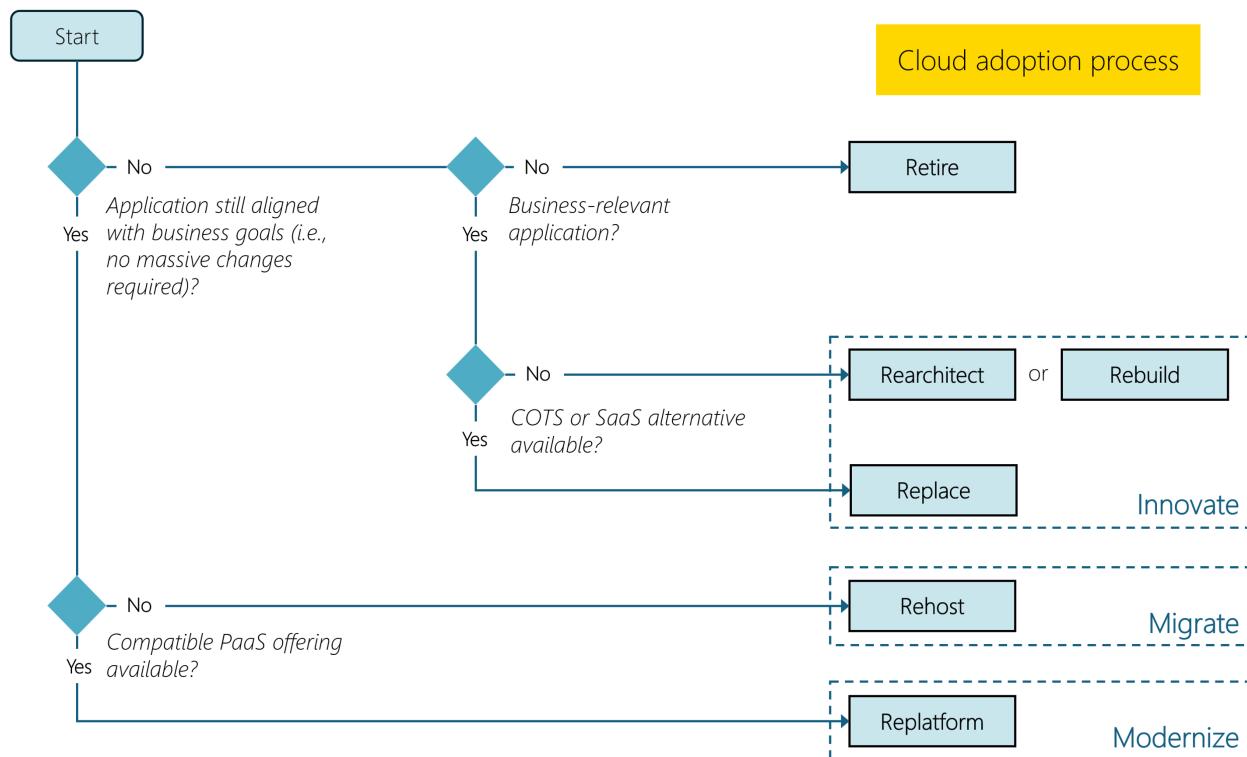
Next steps

- [Overview of the reliability pillar](#)
- [First adoption project](#)

Cloud adoption

Article • 02/22/2024

The Adopt methodology shows you how to migrate, modernize, innovate, and relocate workloads in Azure. These four processes align to different phases in the cloud adoption journey. Each phase has distinct goals, solutions, and benefits. This article provides an overview of each process, so you can find the right guidance.



Migrate

Migration entails moving workloads to the cloud or between clouds. You can choose from various types of migrations. The [the migration guidance](#) helps you choose the migration strategy that best aligns with your objectives.

- **Goals:** Meet business demands, exit an on-premises environment, and align with Azure Well-Architected Framework principles.
- **Solutions:** Adopt cloud solutions based on business needs.
- **Key benefits:** Improve security, reliability, performance, and operations by using managed solutions. Easily integrate new solutions and design patterns without acquiring, managing, or securing hardware.

When you migrate an enterprise-scale workload to the cloud, you shouldn't have to develop new business logic. You can perform migration efficiently by using the

rehosting approach.

For more information, see [Migrate](#).

Modernize

Modernization enhances existing workloads by improving operations, increasing efficiency, maximizing developer velocity, and reducing the total cost of ownership.

- **Goals:** Reduce technical debt, modernize applications, and modernize data platforms.
- **Solutions:** Integrate other services and modify code to meet business goals.
- **Key benefits:** Optimize cost, security, reliability, performance, and operations for increased productivity. You don't need to maintain the underlying infrastructure, so you can focus on your core business.

Generally, application modernization moves a workload toward platform as a service (PaaS) solutions to improve your business at scale. This approach is also known as *replatforming*.

For more information, see [Modernize](#).

Innovate

Innovation is when you adopt cloud-native technologies to create customer-focused solutions that rapidly transform business outcomes.

- **Goals:** Reposition your business, reposition technical solutions, and find innovative data plays.
- **Solutions:** Adopt data and application capabilities to empower adoption and build predictive tools.
- **Key benefits:** Improve predictive analytics, performance, and adaptability.

Application innovation can improve your company's market position and unlock new technical capabilities. AI-powered applications can automate existing business processes and provide new ways to engage with customers. Several strategies are available. Each strategy supports a distinct value and outcome:

- **Rearchitecting:** Extend and optimize the architecture for cloud capabilities and scale.

- **Rebuilding:** Rebuild the code base by using cloud-native technologies.
- **Replacing:** Replace the application with SaaS or low-code solutions.

For more information, see [Innovate](#).

Relocate

Relocation is when you move an Azure workload to a different region in Azure. You can relocate a workload anytime after migration. Relocation evaluations should be a regular part of your workload lifecycle so your workload evolves with your business needs.

- **Goals:** Respond to business changes and expand your global footprint, meet data sovereignty and residency requirements, and provide lower latency to end users.
- **Solutions:** Adopt the location, services, and capabilities of a new Azure region.
- **Key benefits:** Respond to business changes, expand global footprint, meet data sovereignty and residency requirements, provide lower latency to end users.

For more information, see [Relocate](#).

Next steps

Follow the guidance that best meets your goals. If you're still considering cloud adoption, it's help to get a sense of what the cloud adoption journey looks like. The next article outlines the typical cloud adoption journey, showing what workloads you should migrate and the order you should migrate them.

[Cloud adoption journey](#)

Feedback

Was this page helpful?

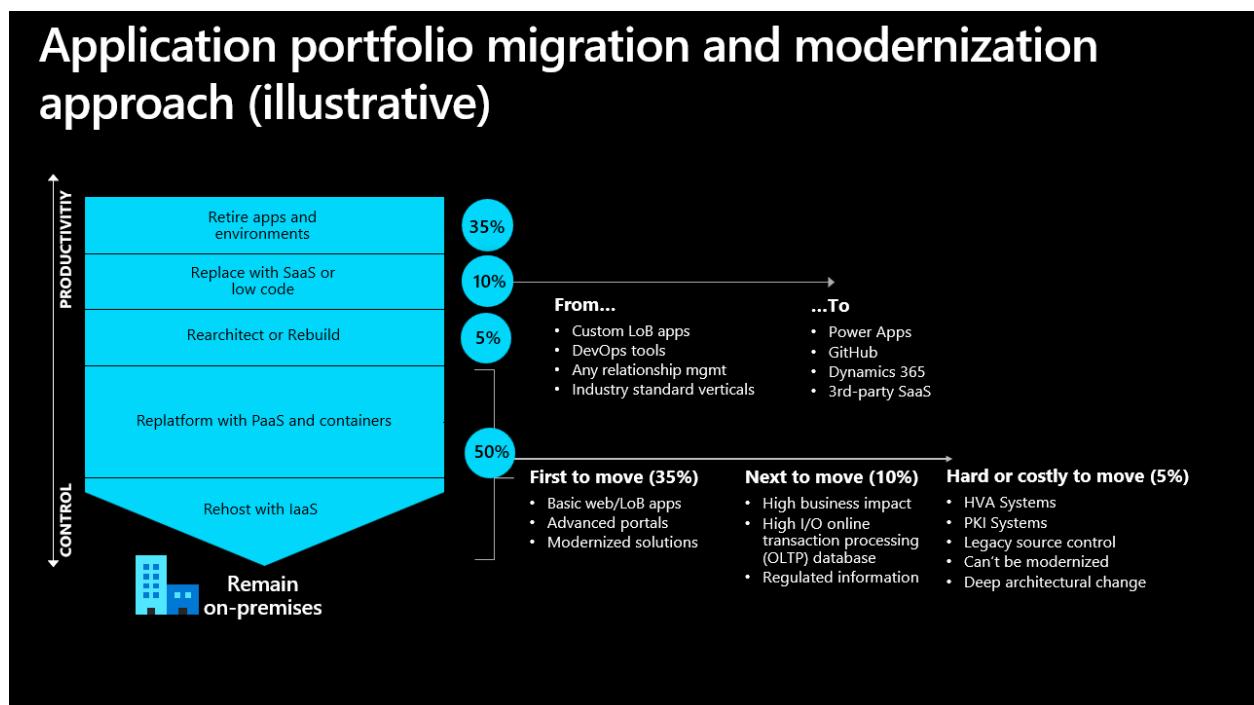
 Yes

 No

Cloud adoption journey

Article • 02/22/2024

Cloud adoption journeys tend to follow similar trajectories. Variations exist, but it can still be helpful to see how others adopt the cloud. Knowing what workloads to tackle first and what to do with them will streamline your cloud adoption journey.



An important consideration in adopting the right cloud solution is the balance of control versus productivity. Infrastructure as a service (IaaS) solutions give you the most control but require more time to maintain. Platform as a service (PaaS) and software as a service (SaaS) solutions transfer management responsibility to Azure and allow your teams to focus on being productive. The balance needed between control and productivity is different for every organization, and it will change over time as your priorities change.

For the initial cloud adoption, a typical organization retires 35% of its applications, replaces 15% of its portfolio, and migrates 50% with only necessary modifications (replatform or rehost).

Retire (35%)

Retire any workloads your organization doesn't need. You need to perform discovery and take inventory to find applications and environments that aren't worth the investment to keep. Cost and time efficiency are the goals of retirement. Your team can focus on the most important assets when you shrink your portfolio before you move it to the cloud.

Replace (10%)

Most organizations replace about 10% of their applications with software as a service (SaaS) and low-code solutions. You can achieve objectives more easily by adopting more productive solutions.

Table 1 - Examples of replacing workloads with SaaS and low-code solutions

[+] [Expand table](#)

From	To
Custom line of business (LOB) applications	Power Apps
DevOps tools	GitHub
Relationship Management	Dynamics 365
Industry verticals	Third-party SaaS ↗

Rearchitect or rebuild (5%)

If you can't effectively replace essential business applications with SaaS or low-code solutions, consider rearchitecting or rebuilding the applications. Although rearchitecting or rebuilding is complex, it's vital for making the most of cloud technology. The main goal is to tailor these applications for the cloud. This approach involves several key aspects:

- *Scalability*: Adapt the application to handle varying demand levels efficiently.
- *Reliability*: Improve the application's ability to operate consistently without failures.
- *Security*: Integrate advanced security measures to protect data and operations in the cloud.

You can also integrate advanced technologies like generative AI at this stage. Integrated solutions can enhance application functionality in significant ways. Examples of AI technology include:

- *Predictive analytics*: Use AI to anticipate customer needs.
- *Process automation*: Employ AI to automate business processes.

By rearchitecting or rebuilding, you exploit the full range of cloud-native capabilities and AI-driven advancements.

Rehost or replatform (50%)

A typical business will migrate about half of its existing workloads. Within these workloads, there are normally three tiers of difficulty. About 35% are easy to move. The next 10% are more difficult because they're more complex or more important, and only the last 5% require extra planning to execute.

There are many migration approaches. Rehosting ("lift-and-shift") and replatforming ("modernize") are the most common, and our recommended approaches for cloud adoption. But it can be difficult to decide which one meets your needs, so we have guidance on deciding which approach is right for you. For more information, see [Migrate or modernize?](#).

First to move (35%)

We recommend picking easy wins for the first workloads to move. This strategy lets you evaluate your adoption plan on easier applications before tackling more complex workloads. As you work, you should document your successes and revise your strategy if needed. Apply these insights to your more complicated moves. Two examples of workloads you could include in your first move are basic web apps and advanced portals.

- **Basic web apps:** We recommend rehosting your basic web applications and waiting to move your more complicated workloads until you've moved your basic web apps. Azure App Service is a flexible application platform that can host most applications. We recommend this solution for basic web applications. For more information, see [Azure App Service](#).
- **Advanced portals:** You should migrate your portals to [Power Apps portals](#) to increase productivity.

Next to move (10%)

You should apply lessons-learned from your first moves to tackle more challenging or more important workloads. We have some examples to give you a sense of the workload types.

- **High business impact:** Workloads that drive revenue or are mission-critical.

- **High input/output (I/O) online transactional processing (OLTP) systems:** These workloads record business transactions and have elevated processing requirements.
- **Regulated information:** These workloads must follow legal and industry standards such as HIPAA, PCI DSS, and others. We recommend using Azure Policy to ensure compliance with these standards. For more information, see [Azure Policy](#).

Hard or costly to move (5%)

Move the most difficult and costly workloads last. The following systems might need more thought to move efficiently.

- **High value asset (HVA):** Disruption or corruption of this workload would disrupt all business operations.
- **Public key infrastructure (PKI) systems:** Workloads that manage x509 digital certificates, network encryption, and authentication.
- **Legacy source control:** Source control systems that aren't easily replaced with GitHub.
- **Can't be modernized:** Legacy or proprietary technology that can't be modernized.
- **Deep architectural change:** Legacy architectures that require a complete redesign of the architecture. Use the [Microsoft Azure Well-Architected Framework](#) rather than the CAF modernize approach.

More resources

- Use the tools available in [Azure Migrate](#) to simplify your journey.
- Join the [migration and modernization program](#) for self-guided digital support and expert-guided deployments.
- Use [FastTrack for Azure](#) to get customized guidance from Azure engineers.

Next steps

Migrate or modernize first?

Feedback

Was this page helpful?

 Yes

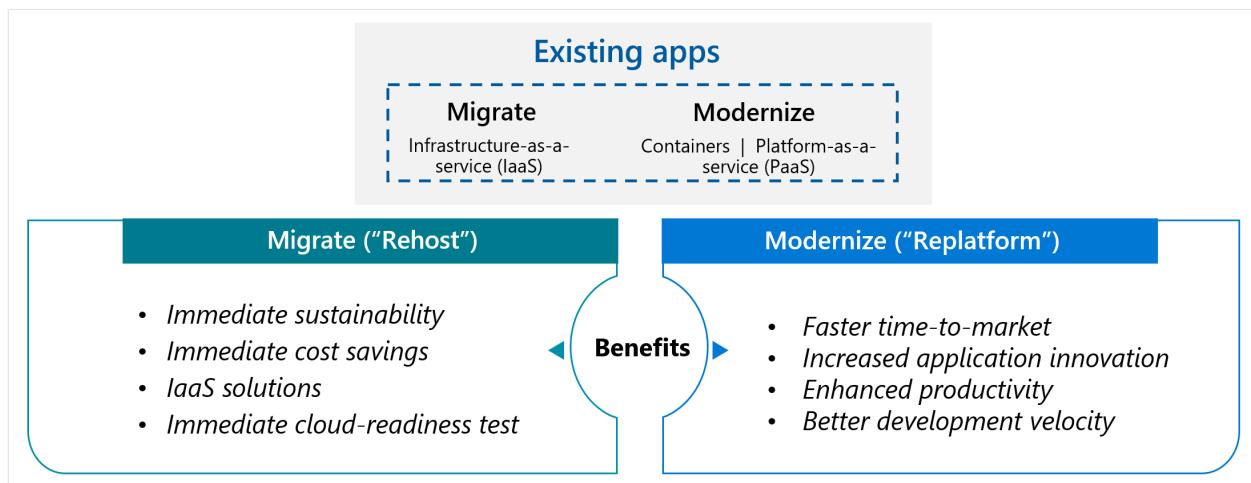
 No

Migrate or modernize first?

Article • 02/22/2024

Figuring out whether you should migrate an on-premises application first ("rehost") or modernize first ("replatform") is a common question. The answer depends on your migration goals, but aligning your goals to an approach can be difficult. To help, we created a short assessment to help you find the right path.

We define each approach and outline their benefits. If the benefits described match your goals, then you've likely found the answer to your question.



Migrate (rehost)

Migrate your applications first using the rehosting approach ("lift-and-shift"). With rehosting, you move an existing application to the cloud as-is and modernize it later. Rehosting has four major benefits:

- **Immediate sustainability:** The lift-and-shift approach is the fastest way to reduce your datacenter footprint.
- **Immediate cost savings:** Using comparable cloud solutions will let you trade capital expense with operational expense. Pay-as-you-go and only pay for what you use.
- **IaaS solutions:** IaaS virtual machines (VMs) provide immediate compatibility with existing on-premises applications. Migrate your workloads to Azure Virtual Machines and modernize while in the cloud. Some on-premises applications can move to an application platform with minimal effort. We recommend Azure App Service as a first option with IaaS solutions able to host all applications.

- **Immediate cloud-readiness test:** Test your migration to ensure your organization has the people and processes in place to adopt the cloud. Migrating a minimum viable product is a great approach to test the cloud-readiness of your organization.

Modernize (replatform)

Modernize your application by using the replatform strategy first. In this approach, you change parts of an application during the migration process. Although it takes a little more work to migrate to the cloud, your applications have better cost and performance efficiency. Modernizing before migration has four benefits:

- **Faster time to market:** Use platform as a service (PaaS) technologies to speed up deployment.
- **Increased application innovation:** PaaS allows developers to focus on business logic and critical data plays.
- **Enhanced productivity:** Adopting PaaS narrows the skills required to push applications to market and increases the productivity of development, security, and operations.
- **Better development velocity:** Switching to managed services will limit the items developers need to focus on and will increase their sprint velocity.

Next steps

Now you have a sense of what to do first. Start migrating or modernizing.

[Learn how to migrate](#)

[Learn how to modernize](#)

Feedback

Was this page helpful?

 Yes

 No

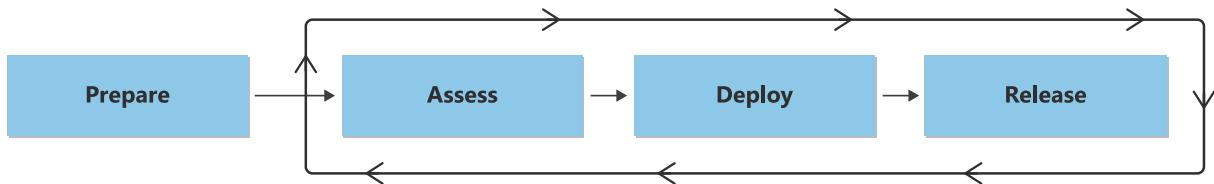
Migrate overview

Article • 04/10/2024

This article describes how to use the Cloud Adoption Framework for Azure to migrate a workload to the cloud and offers guidance for a smooth transition. The migration process includes four stages: prepare, assess, deploy, and release. This approach is vendor-neutral, so you can migrate workloads to any cloud service.

Migration disciplines

A cloud migration typically requires both good preparation and implementation. The Cloud Adoption Framework migration methodology relies on proven guidance to help you prepare for and carry out your migration efforts.



To start a cloud migration, you first need to prepare and fulfill any prerequisites:

- **Prepare:** Focus on organizational alignment. Define clear owners for the migration process and ensure that your teams have the required skills. In this phase, you build on the work from the [strategy](#), [plan](#), and [ready](#) guidance of the Cloud Adoption Framework. This step is critical to the success of your migration effort.

After you fulfill and prepare the prerequisites, you can start the migration process. The migration process is an iterative process that includes the following phases:

- **Assess workloads:** Assess workloads to evaluate cost, modernization, and deployment tooling. This process focuses on validating or challenging assumptions. You make these assumptions during discovery and assessments by looking closely at rationalization options. During this process, you also study user patterns and dependencies closely to ensure the technical success of workloads after migration.
- **Deploy workloads:** After you assess workloads, you replicate or improve the existing workload functionality in the cloud. This replication could involve a *lift and shift* or *rehost* to the cloud. But more commonly, you might modernize many of the assets that support these workloads to capitalize on the benefits of the cloud.

- **Release workloads:** After you replicate workload functionality to the cloud, you can test, optimize, document, and release workloads for ongoing operations. You must review the migrated workloads and hand them off during this process. This step is critical to governance, operations management, and security teams for ongoing workload support.

Cloud migration checklist

[\[+\] Expand table](#)

Migration phase	Activity
<input type="checkbox"/> Prepare	<ul style="list-style-type: none"> • Ready your landing zone for migration • Prepare tools and backlog • Select Azure regions for a migration • Align roles and responsibilities • Learn skills that are relevant to migration projects
<input type="checkbox"/> Assess	<ul style="list-style-type: none"> • Classify workloads • Evaluate workload readiness • Architect workloads
<input type="checkbox"/> Deploy	<ul style="list-style-type: none"> • Deploy supporting services • Remediate assets • Replicate assets • Prepare for management • Test the migration
<input type="checkbox"/> Release	<ul style="list-style-type: none"> • Begin change communication • Conduct business testing • Complete the migration • Optimize costs after the migration • Conduct retrospectives

For more details, see the articles for each phase.

Audience

The Migrate methodology addresses various roles and functions, for example:

- **Business decision makers:** Understand the motivations for migrations. Make informed decisions that relate to your overall budget and cloud investments. Learn about relevant skills that your organization needs to migrate workloads and how to get assistance from partners.

- **IT decision makers:** Understand the considerations for migrating into multiple regions or multiple datacenters. Explore the recommended migration tools and skills required to migrate.
- **Platform owners or platform architects:** Learn how to prepare your Azure landing zones for migration. Also explore other technical preparations for your Azure estate that you might need to implement before you initiate a migration project.
- **Cloud engineers or cloud architects:** Assess existing workloads and solutions, deploy required infrastructure in the cloud, and release workloads into production.

Next step

Familiarize yourself with the prepare discipline to get started.

Prepare

Feedback

Was this page helpful?

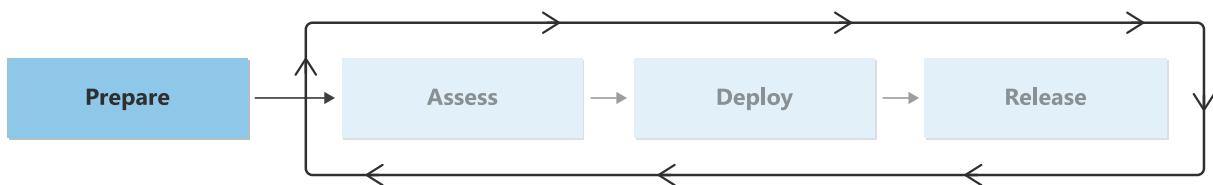
 Yes

 No

Migration preparation checklist

Article • 04/10/2024

Before you plan individual workload migrations, you must ready your organization and cloud resources to support the migration. Consider the decisions you need to make and the processes you need to incorporate to ready your environment. Some of these considerations are covered earlier in the Cloud Adoption Framework for Azure, and those decisions can affect future migration activities.



The following section provides guidance about creating a secure foundation and plan for your migrations.

It's important to prepare. If you fail to prepare for migrations, you might run into setbacks if you have to retroactively incorporate changes. If the process becomes too intensive, you might even have to abandon the migration and resume it at a later date, after tasks are completed. To avoid this scenario, make sure that you have a plan for addressing all the readiness activities.

Checklist

The following table provides an overview of the articles that describe the activities for migration preparation. It also shows the roles that are responsible for each activity.

[\[\] Expand table](#)

Activity	Description	Responsible roles
Prepare your landing zone for migration	Learn the components that your landing zone needs to support migration activities.	<ul style="list-style-type: none">Landing zone architect
Prepare tools and an initial migration backlog	Learn how to prepare the right tools and build a migration backlog from your digital estate planning.	<ul style="list-style-type: none">Project sponsorMigration architectProject manager
Select Azure regions for a migration	Learn how to plan for and select the Azure cloud regions for your workload migration to Azure. Find	<ul style="list-style-type: none">Migration architect

Activity	Description	Responsible roles
	suggested actions for assessment, migration, and other processes.	<ul style="list-style-type: none"> • Landing zone architect • Cloud operations manager
Align roles and responsibilities	Learn about the roles and functions that you need for a migration project.	<ul style="list-style-type: none"> • Project manager • Stakeholder
Incorporate skills readiness for migration	Find out about the resources and training that your team needs to build essential migration skills.	<ul style="list-style-type: none"> • Project manager • Organizational change manager

Next steps

Understand how to prepare your landing zone for migrations.

[Prepare your landing zone](#)

Feedback

Was this page helpful?

 Yes

 No

Prepare your landing zone for migration

Article • 04/10/2024

This article describes how to get your [Azure landing zone](#) ready for a migration. It also lists the major tasks you must perform to ensure that configurations are in place for your migration project.

Regardless of which [Azure landing zone reference implementation](#) you used, you must perform some tasks to prepare your landing zone for a successful migration project.

If you didn't use an Azure landing zone reference implementation, you still need to perform the steps in this article. However, you might have prerequisite tasks to perform first, or you might have to adapt specific recommendations to your design.

This article describes the tasks you must perform for your existing Azure landing zone after it's deployed. Some tasks focus on automated deployments. It's noted if a task isn't relevant for manually deployed and managed environments.

Establish hybrid connectivity

During an Azure landing zone deployment, you can deploy a Connectivity subscription with a hub virtual network and network gateways, such as Azure VPN gateways, Azure ExpressRoute gateways, or both. After your Azure landing zone deployment, you must still configure hybrid connectivity from these gateways to connect to your existing datacenter appliances or your ExpressRoute circuit.

In the ready phase, you planned for your [connectivity to Azure](#). Use this plan to determine the connections that you need to incorporate. For example, if you use ExpressRoute, you must work with your provider to establish your ExpressRoute circuit.

To get technical guidance for specific scenarios, see:

- [Create a VPN connection from your Azure VPN gateway](#).
- [Create an ExpressRoute circuit](#).
- [Create an ExpressRoute connection from your ExpressRoute gateway to your circuit](#).
- [Manage Azure Virtual WAN gateway settings](#).

Note

For additional guidance, also refer to your provider's specific documentation.

If you establish your hybrid connectivity to Azure via a third-party network virtual appliance (NVA) that's deployed in your virtual network, review their specific guidance and our [general guidance for highly available NVAs](#).

Prepare identity

During your Azure landing zone deployment, you should also deploy a supporting architecture for your identity platform. You might have a dedicated identity subscription or resource groups and a virtual network or subnets for the virtual machines (VMs) that you use for identity. However, you must deploy the identity resources after the Azure landing zone deployment.

The following sections provide guidance related to Active Directory. If you use a different identity provider for authentication and authorizations, you must follow their guidance on extending your identity to Azure.

Before you implement this guidance, review the [Active Directory and hybrid identity](#) decisions that you made when you planned for your landing zone.

You should also review your [identity baseline](#) from the governance phase to determine if you need to make changes in Microsoft Entra ID.

Extend Active Directory domain controllers

In most migration scenarios, the workloads that you migrate to Azure are already joined to an existing Active Directory domain. Microsoft Entra ID offers solutions for modernizing identity management, even for VM workloads, but it can disrupt migration. Rearchitecting identity usage for workloads is often performed during modernization or innovation initiatives.

As a result, you need to deploy domain controllers to Azure inside the identity network area that you deployed. After you deploy VMs, you must follow your normal domain controller promotion process to add them to the domain. This process might include creating additional sites to support your replication topology.

For a common architecture pattern to deploy these resources, see [Deploy Active Directory Domain Services \(AD DS\) in an Azure virtual network](#).

If you implement the [enterprise-scale architecture for small enterprises](#), the AD DS servers are often in a subnet in the hub. If you implement the [enterprise-scale hub-and-spoke architecture](#) or the [enterprise-scale Virtual WAN architecture](#), the servers are often in their dedicated virtual network.

Microsoft Entra Connect

Many organizations already have Microsoft Entra Connect to populate Microsoft 365 services, like Exchange Online. If your organization doesn't have Microsoft Entra Connect, you might need to [install it](#) and deploy it after your landing zone deployment so you can replicate identities.

Enable hybrid DNS

Most organizations need to be able to resolve Domain Name System (DNS) requests for namespaces that are a part of existing environments. These namespaces often require integration with Active Directory servers. And resources in the existing environment must be able to resolve resources in Azure.

To enable these functions, you need to configure DNS services to support common flows. You can use Azure landing zones to deploy many of the resources you need. For additional tasks to review and prepare for, see [DNS resolution in Azure](#).

Custom DNS resolution

If you use Active Directory for your DNS resolver or if you deploy a third-party solution, you must deploy VMs. You can use these VMs as your DNS servers if your domain controllers are deployed to your Identity subscription and network spoke. Otherwise, you must deploy and configure the VMs to house these services.

After you deploy the VMs, you must integrate them into your existing DNS platform so they can perform lookups against your existing namespaces. For Active Directory DNS servers, this integration is automatic.

You can also use [Azure DNS Private Resolver](#), but this service isn't deployed as part of your Azure landing zone deployment.

If your design uses private DNS zones, plan accordingly. For example, if you use private DNS zones with private endpoints, see [Specify DNS servers](#). Private DNS zones are deployed as part of your landing zone. If you also use private endpoints to perform modernization efforts, you should have an additional configuration for them.

Azure Firewall DNS proxy

You can configure Azure Firewall as a [DNS proxy](#). Azure Firewall can receive traffic and forward it to an Azure resolver or your DNS servers. This configuration can allow

lookups to be performed from on-premises to Azure, but they can't be conditionally forwarded back to on-premises DNS servers.

If you need hybrid DNS resolution, you can configure the Azure Firewall DNS proxy to forward traffic to your custom DNS servers, such as your domain controllers.

This step is optional, but it has several benefits. It reduces configuration changes later if you change DNS services and enables fully qualified domain name (FQDN) rules in Azure Firewall.

Configure custom virtual network DNS servers

After you complete the preceding activities, you can configure the DNS servers for your Azure virtual networks to the custom servers that you use.

For more information, see [Azure Firewall DNS settings](#).

Configure hub firewall

If you deployed a firewall in your hub network, there are a few considerations that you should address so you're ready to migrate workloads. If you don't address these considerations early in your deployment, you might run into routing and network access problems.

As part of performing these activities, review the [networking design area](#), especially the [network security guidance](#).

If you deploy a third-party NVA as your firewall, review the vendor's guidance and our [general guidance for highly available NVAs](#).

Deploy standard rule sets

If you use an Azure firewall, all firewall traffic is blocked until you add explicit allow rules. Many other NVA firewalls work similarly. Traffic is denied until you define rules that specify the traffic that's permitted.

You should add individual rules and rule collections based on workload needs. But you should also plan to have standard rules, such as access to Active Directory or other identity and management solutions, that apply to all enabled workloads.

Routing

Azure provides routing for the following scenarios with no additional configuration:

- Routing between resources in the same virtual network
- Routing between resources in peered virtual networks
- Routing between resources and a virtual network gateway, either in its own virtual network or in a peered virtual network that's configured to use the gateway

Two common routing scenarios need additional configuration. Both scenarios have route tables assigned to subnets to shape routing. For more information about Azure routing and custom routes, see [Virtual network traffic routing](#).

Inter-spoke routing

For the [network design area](#), many organizations use a [hub-spoke network topology](#).

You need routes that transfer traffic from one spoke to another. For efficiency and simplicity, use the default route (`0.0.0.0/0`) to your firewall. With this route in place, traffic to any unknown location goes to the firewall, which inspects the traffic and applies your firewall rules.

If you want to allow for internet egress, you can also assign another route for your private IP space to the firewall, such as `10.0.0.0/8`. This configuration doesn't override more specific routes. But you can use it as a simple route so inter-spoke traffic can properly route.

For more information on spoke-to-spoke networking, see [Patterns and topologies for inter-spoke communication](#).

Routing from the gateway subnet

If you use virtual networks for your hub, you need to plan how to handle the inspection of traffic that comes from your gateways.

If you intend to inspect traffic, you need two configurations:

- In your Connectivity subscription, you need to create a route table and link it to the gateway subnet. The gateway subnet needs a route for every spoke network you intend to attach, with a next hop of your firewall's IP address.
- In each of your landing zone subscriptions, you need to create a route table and link it to each subnet. Disable Border Gateway Protocol (BGP) propagation on the route tables.

For more information about custom and Azure-defined routes, see [Azure virtual network traffic routing](#).

If you intend to inspect traffic to private endpoints, enable the appropriate routing network policy on the subnet where the private endpoints are hosted. For more information, see [Manage network policies for private endpoints](#).

If you don't intend to inspect traffic, no changes are needed. However, if you add route tables to your spoke network subnets, enable BGP propagation so traffic can route back to your gateway.

Configure monitoring and management

As part of deploying your landing zone, you have provisioned policies that enroll your resources in Azure Monitor Logs. But you must also create alerts for your landing zone resources.

To implement alerts, you can deploy the [Azure Monitor baseline for landing zones](#). Use this deployment to get alerts based on common scenarios for landing zone management, such as connectivity resources and service health.

You can also deploy your own custom alerting for resources if your needs deviate from what's in the baseline.

Prepare your landing zone for sovereign workload migrations

If you need to address sovereignty requirements, you can evaluate if [Microsoft Cloud for Sovereignty](#) fits your requirements. Microsoft Cloud for Sovereignty provides an additional layer of policy and auditing capabilities that address individual public sector and government customer needs.

You can enable these capabilities by deploying the [sovereign landing zone](#). The architecture of the sovereign landing zone aligns with the recommended [Azure landing zone](#) designs.

Microsoft Cloud for Sovereignty policy portfolio

By using Azure policy, you can enable centralized control across Azure resources to enforce specific configurations. You can assign the [Microsoft Cloud for Sovereignty](#)

[policy initiatives](#) to your landing zones to make sure you adhere to local policies and regulatory requirements in your country/region.

If those policy initiatives are not yet assigned to your sovereign landing zone deployment, consider assigning the initiatives that correspond to your regulatory requirements.

Enable subscription vending

This section applies to organizations that want to automate their subscription provisioning process. If you manually manage your landing zone and subscription creation, you should establish your own process for creating subscriptions.

When you begin migrating, you must create subscriptions for your workloads. Enable [subscription vending](#) to automate and accelerate this process. When subscription vending is established, you should be able to create subscriptions quickly.

Prepare for Microsoft Defender for Cloud

When you deploy your landing zone, you also set policies to enable [Defender for Cloud](#) for your Azure subscriptions. Defender for Cloud provides security posture recommendations in its [secure score](#), which evaluates deployed resources against the Microsoft security baseline.

You don't need to implement additional technical configurations, but you should review the recommendations and design a plan to [improve your security posture](#) as you migrate resources. When you begin migrating resources into Azure, you should be ready to [implement security improvements](#) as part of your migration optimization.

Related resources

Consider these additional resources to prepare for migration:

- [Prepare an initial corporate policy that's defined and well understood](#)
- [Create an adequate plan for Azure billing](#)
- [Ensure that you have proper organizational alignment and a plan to manage it](#)
- [Develop naming and tagging standards](#)

Next steps

Feedback

Was this page helpful?

 Yes

 No

Prepare tools and initial migration backlog

Article • 04/10/2024

To implement your migration, you need the right tools and a comprehensive backlog of workloads to migrate. This article provides guidance on how to prepare for your migration by defining the necessary tools and building an initial migration backlog.

Prepare migration tools

To successfully complete your migration, you need specific tools for assessing, replicating, and tracking your workloads through iterations, including remediation activities.

Various migration tools are available. Many are either native to the Azure platform or are already commonly available.

Here's a list of common tools or offerings that you need for a successful migration project:

[+] Expand table

Tool type	Functionality	Tool
Discovery and assessment	Performs automated discovery and assessments of your environment. Identifies blockers for migration and identifies dependencies between servers.	Azure Migrate
Replication	Replicates data state between your on-premises source and a cloud staging environment. Used to hydrate and migrate the resources.	Azure Migrate
Tracking	Used to organize project activities, such as to group servers into workloads, track remediation activities, and provide the status of a workload migration.	Azure DevOps , Excel, and Microsoft Project
Migration guide	Helps you identify which migration feature to use for Azure Migrate. The Migration Execution Guide is a project resource that can guide you step by step through the decisions and the implementation of your migration.	Migration Execution Guide ↗

Although you can use other tools instead of Azure Migrate, we recommend that you use a native offering unless there's an identified reason. The native offering Azure Migrate is

built to work seamlessly with the Azure platform and is continuously updated to support the latest features and capabilities.

Note

If you use an existing tool to replicate a workload to Azure, changing tools during this process can disrupt and decrease performance. In this scenario, continue to use the existing tool. Later, you can run the migration promotion like a disaster recovery failover scenario.

Initial migration backlog

The following sections describe the prerequisite activities that you should perform to build an initial migration backlog.

The cloud strategy team is accountable for the care and maintenance of the digital estate. However, addressing the backlog generated from mapping a digital estate falls under the shared responsibility of all roles involved in the migration process. The cloud strategy team and the cloud adoption team should review and understand the migration backlog before the teams begin to plan individual workload activities. During review, members of both teams must gain sufficient knowledge to articulate the following key points about the migration backlog.

Business outcomes and metrics

Every member of the team should understand the intended business outcomes. Migrations take time. It's easy for team members to become distracted by urgent but less strategic activities at different stages of a migration. Establishing and reinforcing intended outcomes helps team members understand the prioritization and relative importance of migration activities so that they make better decisions over time.

Tracking migration progress is equally important both to the migration team's motivation and to continued stakeholder support. Track progress through migration KPIs and by monitoring metrics. Regardless of how you track the effort, it's important for the team to be aware of key metrics so that it can evaluate performance during subsequent iterations.

Business priorities

Sometimes, prioritizing one workload over another might not seem logical or even beneficial to the cloud adoption team. Understanding the business priorities that drive workload prioritization decisions can help the team maintain critical motivation. It also helps the team make a stronger contribution during the prioritization decision-making process.

Core assumptions

[Digital estate rationalization](#) discusses the agility and time-saving impact of working from basic assumptions when you evaluate a digital estate. To fully realize those values, the cloud adoption team needs to understand the assumptions, and the reasons that assumptions were established. That knowledge better equips the team to challenge assumptions for effectiveness and savings.

Capture the backlog

Capture the backlog in a location that you can share with all members of the cloud adoption team. From a shared location, different team members can align their knowledge and work to the backlog, and you can keep the backlog current throughout the migration process.

You can both use tools that are familiar in your organization, and build on the tools that you use to complete your digital estate rationalization.

If you're looking for prebuilt templates, the [Migration Execution Guide](#) has spreadsheet templates that can help you organize your backlog.

It's important to associate workloads with servers, so you can track a workload itself through individual server migrations in the backlog. You can also use the backlog to showcase dependencies between workloads as you complete the assessment. When you [remediate assets](#) and complete testing, you merge the backlog with a remediation plan.

The backlog is used throughout the migration process. Maintaining the backlog is critical.

Plan the backlog for multiple datacenters

Before you begin your migration, you should create epics in the project management tool for each datacenter that you migrate. In a datacenter epic, you can group associated work so that you can track the status of each datacenter location.

If you don't use epics to manage your migration, you can use top-level goals or groupings for datacenters. The key is that you can filter, organize, and track each datacenter as a separate location.

It's important to understand the business outcomes and motivations for your migration. Use those motivations to prioritize the list of epics (or datacenters). For example, if the intention to migrate from an on-premises datacenter before the end of your current lease drives your migration, prioritize epics by using lease renewal dates.

Within each epic, manage workloads that you assess and migrate as features. Manage each asset within that workload as a user story. Tasks represent work to assess, migrate, optimize, promote, secure, and manage each asset.

A sprint or iteration is a series of tasks that are required to migrate the assets and user stories that the cloud adoption team commits to. A sprint typically is a segment of time, like a fiscal quarter or a calendar month. A release represents one or more workloads or features that are promoted to production.

Next step

[Plan for resilience](#)

Feedback

Was this page helpful?

 Yes

 No

Select Azure regions for a migration

Article • 04/10/2024

When you migrate an existing environment to Azure, you need to select an Azure region or set of regions to host the migrated components. Region selection involves the following high-level steps:

- Review the [core Azure region selection guidance](#) to understand how to select Azure regions that meet your requirements.
- **Inventory and document the current state** of your environment.
- **Implement a general approach** to your migration, including whether to run in a single region, use multiple availability zones, or use multiple regions.
- **Assess process changes** that might be required.
- **Plan a migration process.**
- **Optimize and promote process changes.**

This article provides guidance on how to choose Azure regions that meet your migration needs. If you haven't already, you might need to extend your [landing zone regions](#) to support multi-region approaches.

ⓘ Note

This article covers considerations that are specific to workload migrations. You should also understand general principles for selecting Azure regions for any organization or workload. For more information, see [Select Azure regions](#).

Document your scenario complexity

Determine whether your scenario requires documentation and process alignment. The following approach can help you assess potential challenges and establish a general course of action:

- **Consider a more robust readiness and governance implementation.**
- **Inventory the affected geographies.** Compile a list of the affected countries or regions.
- **Document the user base.** Will the cloud migration affect employees, partners, or customers in the identified country or region?
- **Document datacenters and assets.** Does the migration effort include any assets in the identified country or region?
- **Document regional product version availability and failover requirements.**

- **Document your resiliency requirements** to determine whether availability zones are required. Typically, you consider resiliency requirements for the whole scenario, not for individual regions.
- **Document your sovereignty requirements and data residency requirements.** Workloads that have specific sovereignty or data residency requirements might influence your choice of Azure regions.

Throughout the migration process, consider how to align changes across your various scenarios and inventories. The following table shows an example of how to document various scenarios.

[\[+\] Expand table](#)

Region	Country/region	Local employees	Local external users	Local datacenters or assets	Data sovereignty requirements
North America	United States	Yes	Partners and customers	Yes	No
North America	Canada	No	Customers	Yes	Yes
Europe	Germany	Yes	Partners and customers	No - network only	Yes
Asia Pacific	South Korea	Yes	Partners	Yes	No

Why is the location of users relevant?

Organizations that support users in multiple countries or regions develop technical solutions that address user traffic. In some cases, solutions involve localization of assets. In other scenarios, the organization might choose to implement global wide area network (WAN) solutions to address disparate user bases through network-focused solutions. In either case, the usage profiles of disparate users can affect the migration strategy.

For example, if an organization supports employees, partners, and customers in Germany but doesn't currently have datacenters in Germany, the organization probably implements a *leased-line* solution. This type of solution routes traffic to datacenters in other countries or regions. This existing routing presents a significant risk to the perceived performance of migrated applications. Injecting more hops in an established

and tuned global WAN can create the perception of underperforming applications after migration. Finding and fixing those issues can add significant delays to a project.

Each of the following sections includes guidance for addressing this complexity across prerequisites and processes of assessing, migrating, and optimizing. Understanding user profiles in each country or region is critical for properly managing this complexity.

Why is the location of datacenters relevant?

The location of existing datacenters can affect a migration strategy. Consider the following factors:

Architecture decisions: One of the first steps in migration strategy design is to determine the target region. The location of existing assets often influences this determination. Also, the availability of cloud services and the unit cost of those services can vary between regions. Data residency requirements, including sovereignty requirements, might also influence the architecture decision. Understanding where current and future assets are located affects architecture decisions and can influence budget estimates.

Datacenter dependencies: In the table in the [Document your scenario complexity](#) section, the example scenarios show that you probably need to plan for dependencies between various global datacenters. Organizations that operate on this scale might not document or clearly understand these dependencies. Your organization's approach to evaluating user profiles helps you identify some of these dependencies in your organization. Your team should also explore more assessment steps that can help mitigate the risks and complexities that arise from dependencies.

Implement a general approach

The following approach uses a data-driven model to address global migration complexities. If the migration scope includes multiple regions, the cloud adoption team should evaluate the following readiness considerations:

- **Determine whether you can meet your business requirements:** Use multiple availability zones to determine requirements for high availability, resiliency, performance, and cost. If these requirements aren't met, consider whether you need a multi-region approach.
- **Evaluate data sovereignty:** Data sovereignty can require localization of some assets, but many assets aren't governed by those compliance constraints. Services like logging, reporting, network routing, identity, and other central IT services

might be eligible to be hosted as shared services across multiple subscriptions or regions. Evaluate data sovereignty by using a shared service model for those services. For an outline of this approach, see the [reference architecture for a hub-spoke topology with shared services](#).

- **Ensure that your environment scales:** If you deploy multiple instances of similar environments, you can establish a dedicated team for the environment's migrations to help create consistency, improve governance, and accelerate deployment. The [governance guide for complex enterprises](#) establishes an approach that creates an environment that scales across multiple regions.

Data-driven prerequisites

When your team is comfortable with the baseline approach and readiness is aligned, consider these data-driven prerequisites:

- **Complete general discovery:** Complete the table in [Document complexity](#) to evaluate the complexity of your cloud adoption strategy.
- **Analyze user profiles for each affected country or region:** It's important to understand general user routing early in the migration process. Changing global lease lines and adding connections like Azure ExpressRoute to a cloud datacenter can result in months of networking delays. Address user routing as early in the process as possible.
- **Complete an initial digital estate rationalization:** If you introduce complexity into a migration strategy, complete an initial digital estate rationalization. For more information, see [What is a digital estate?](#).
- **Use tagging for digital estate requirements:** Establish tagging policies to identify any workload that's affected by data sovereignty requirements. Ensure that required tags begin in digital estate rationalization and carry through to migrated assets.
- **Evaluate a hub-spoke model:** Distributed systems often share common dependencies. You can often address shared dependencies by implementing a hub-spoke model. Although implementing a hub-spoke model is out of scope for the migration process, flag the model for consideration during future iterations of the [ready processes](#).
- **Prioritize the migration backlog:** If you require network changes to support production deployment of a workload that supports multiple regions, the cloud strategy team should track and manage escalations that result from the network

changes. This higher level of executive support helps accelerate the change by freeing the strategy team to reprioritize the backlog and ensure that network changes don't block global workloads. Prioritize global workloads only when network changes are finished.

These prerequisites help establish processes that can address complexity during execution of the migration strategy.

Assess process changes

If your migration scenario involves global asset and user-base complexities, add key activities to assess your migration candidates. These activities produce data to help you clarify obstacles and outcomes for global users and assets.

Suggested actions during the assess process

Evaluate cross-datacenter dependencies: The [dependency analysis tools in Azure Migrate](#) can help you pinpoint dependencies. Use these tools before you begin migration. If your scenario involves global complexity, evaluating dependencies is a necessary step in the assess process. You can use [dependency grouping](#) to visualize dependencies and identify the IP addresses and ports of any assets that are required to support the workload.

Important

- You need a subject matter expert (SME) who understands asset placement and IP address schemas to identify assets that are located in a secondary datacenter.
- Evaluate downstream dependencies and clients in the visualization to understand bidirectional dependencies.

Identify global user impact: The output from the prerequisite user-profile analysis should identify any workload that's affected by global user profiles. If a migration candidate is in the list of affected workloads, the migration architect should consult networking and operations SMEs. These experts help validate network routing and performance expectations. At a minimum, the architecture should include an ExpressRoute connection between the closest network operations center and Azure. The [reference architecture for ExpressRoute connections](#) can help you configure the necessary network connections.

Design for compliance: The output from the prerequisite user-profile analysis should also identify any workload that's affected by data sovereignty requirements. During the architecture activities of the assess process, the assigned architect should consult compliance SMEs. These experts can help the architect understand any requirements for migration and deployment across multiple regions. Those requirements significantly affect design strategies. The following reference architectures can help with the design:

- Zone-redundant web applications
- Multi-region web applications
- Multi-region n-tier applications
- Workload templates for a sovereign landing zone

 **Warning**

If you use the reference architecture for ExpressRoute or the reference architectures for applications, you might need to exclude specific data elements from replication processes to meet data sovereignty requirements. The task of excluding specific data elements adds a step to the promotion process.

Migration process changes

If you migrate an application that must be deployed to multiple regions, the cloud adoption team must account for a few more considerations. The design of Azure Site Recovery vaults and configuration and process servers are two of those considerations. Two other considerations are network-bandwidth designs and data synchronization.

Suggested actions during the migration process

Site Recovery vault design: Site Recovery is the suggested tool for cloud-native replication and synchronization of digital assets to Azure. Site Recovery replicates data about each asset to a Site Recovery vault. This vault is bound to a specific subscription in a specific region and Azure datacenter. If you replicate assets to a second region, you might also need a second Site Recovery vault.

Configuration and process server design: Site Recovery works with a local instance of a configuration and process server that's bound to a single Site Recovery vault. When you use this configuration, you might need to install second instances of these servers in the source datacenter to facilitate replication.

Network bandwidth design: During replication and ongoing synchronization, you move binary data over the network from the source datacenter to the Site Recovery vault in

the target Azure datacenter. The replication and synchronization process consumes bandwidth. Duplicating the workload to a second region doubles the amount of consumed bandwidth.

In some scenarios, bandwidth is limited. In others, a workload involves substantial configuration or data drift. In these cases, replicating data to a second region can interfere with the time that it takes to complete the migration. More importantly, these constraints can affect the experience of users or applications that still depend on the bandwidth that was available in the source datacenter.

Data synchronization: The largest bandwidth drain often comes from synchronizing the data platform. If you deploy across multiple availability zones, you might be able to use zone-redundant data services that automatically synchronize your data across multiple availability zones. Deployment across multiple regions often requires data synchronization to keep applications aligned. This approach is defined in the reference architectures for [multi-region web applications](#) and [multi-region n-tier applications](#).

If keeping applications synchronized is the operational state you want for your applications, you might want to synchronize the source data platform with each cloud platform. Perform this sync before you migrate the application and middle-tier assets.

Azure-to-Azure disaster recovery: An alternative option might further reduce complexity. If you use a two-step deployment to meet timeline and data-synchronization needs, [Azure-to-Azure disaster recovery](#) might be an acceptable solution. In this scenario, you migrate the workload to the first Azure datacenter by using a single Site Recovery vault and configuration and process server design. After you test the workload, you can recover the workload to a second Azure datacenter from the migrated assets.

This approach reduces the effect on resources in the source datacenter. Azure-to-Azure disaster recovery also takes advantage of fast transfer speeds and high bandwidth limits between Azure datacenters.

Note

The Azure-to-Azure disaster recovery approach can increase short-term migration costs through higher egress bandwidth charges.

Release process changes

As you address global complexity during optimization and promotion, you might require identical efforts in each region that you deploy to. If you use a single region, you might still need to replicate business testing and business change plans.

Suggested actions during the release process

Pretest optimization: Initial automation testing can identify potential optimization opportunities, as with any migration effort. For global workloads, independently test the workload in each region. Minor configuration changes in your network or in the chosen Azure datacenter can affect performance.

Business change plans: Create a business change plan for any complex migration scenario. A business change plan helps ensure clear communication about changes to business processes and user experiences. The plan also helps ensure clear communication about the timing of efforts that are required to integrate changes. In a global migration effort, the plan should include considerations for users in each affected geography.

Business testing: Each region might also require business testing. Business testing helps ensure adequate performance and adherence to modified network routing patterns.

Promotion flights: Promotion often happens as a single activity, and production traffic is immediately rerouted to the migrated workloads. In a global release effort, you should deliver promotion in predefined collections of users called *flights*. Promotion flights give the cloud strategy team and cloud adoption team an opportunity to observe performance and improve support for users in each region. You can control promotion flights at the networking level. Specifically, you can change the routing of specific IP ranges from the source workload assets to the newly migrated assets. After you migrate a specified collection of users, you can reroute the next group.

Flight optimization: One benefit of promotion flights is that they give you deeper observations and an opportunity to optimize the deployed assets. After the first flight successfully uses production for a brief time, you can refine the migrated assets when IT operation procedures support it.

Feedback

Was this page helpful?



Align roles and responsibilities

Article • 04/10/2024

Understanding an organization's culture and datacenter management is vital for Azure migration success. Centralized IT teams with clear roles ease the process, but larger or compliance-bound enterprises face nuanced challenges that can hinder progress.

The Azure Cloud Adoption Framework underscores the role of [organizational alignment](#) in migration, advocating for cross-departmental collaboration to fulfill key functions.

In this article, you learn about:

- Migration-specific roles that align with the [cloud strategy](#) and [cloud adoption](#) functions.
- Supporting roles that you might need for other functions during the migration process, for example Landing zone architects and workload architects.
- How to identify relevant experts or owners for roles in your migration projects.
- A responsibility matrix to help understand which role is responsible for what part of a migration project.

Tip

The roles mentioned might not match specific job titles or require dedicated team members. Often, one person can cover multiple roles, or several team members can share responsibilities. This list outlines common responsibilities but isn't a staffing guide. The key is ensuring that these responsibilities are met within your organization.

Cloud strategy function roles

To make sure that you have the necessary commitment and organization for your migration project, you need the following roles for the [cloud strategy function](#). The following table describes the cloud strategy function roles and their responsibilities:

 Expand table

Role	Responsibilities
Project Sponsor	Defines the scope of the migration to determine what resources are moved and the benefit of moving each resource. Provides decision-making ownership for migration tooling purchases, for the overall workload architecture, and for release activities.
Project Manager	Drives a project plan for the migration scope. Drives testing processes. Organizes status updates to stakeholders.
Organizational Change Manager	Helps the project team communicate changes to the organization. Works with different functions to make sure that the right team members are involved and that the correct organizational changes occur to support the migration.
Licensing Specialist	Provides licensing insight and financial operations management to ensure that the project is properly licensed and uses existing licensed resources.
Workload Business Owner	Provides decision-making ownership for the workload assessment, architecture, and migration processes. Acts as an owner for the business value of the workload in Azure.

Cloud adoption function roles

During your migration to Azure, the [cloud adoption function](#) performs most of the technical execution. For this function, plan to have the roles that are described in the following table:

 Expand table

Role	Responsibilities
Migration Architect	Oversees the technical decision making for the workloads, such as migration wave planning and all migration processes.
Migration Engineer	Executes tasks that are identified as part of the project.

Supporting roles for other functions

The next table describes supporting roles that you might need for other functions:

Role	Responsibilities
Landing Zone Architect	Provides support for migrating workloads to a landing zone. Helps remediate any issues with platform services in the landing zone. For more information, see Cloud platform functions .
Cloud Operations Manager	Provides support for onboarding migrating workloads to the management platform to ensure that proper management is in place for the workloads when they migrate. For more information, see Cloud operations functions .
Workload Architect	Provides architectural guidance and decision making for the design of the migrating workload. For each workload, you might need a specific subject matter expert to fulfill multiple instances of this role. For more information, see Central IT functions .
User Acceptance Tester	Tests individual workloads. You might have multiple instances of this role per workload to provide feedback for user acceptance testing (UAT). For more information, see Central IT functions .

Identify experts or owners for roles

It can be difficult to identify the correct resources for some of these roles, such as for Workload Architect and Workload Business Owner. If a workload is in maintenance for a long period and without frequent changes, you might find limited ownership information and technical expertise to support a function. For example, in digital estate planning, sometimes servers aren't mapped to a specific workload, so it can be unclear who has ownership of them.

Here are some recommendations to identify roles:

- Historical data:** Use your configuration management database or ticketing system to identify any historical items that indicate who requests maintenance or who communicates about the server or workload.
- Sign-in logs:** Look for the users who were most recently logged in on the servers in the workload. Although this approach might not identify an owner, recent users can give you context for the server.
- Dependency analysis:** Use dependency analysis tools to identify who most frequently connects to the functions that are hosted on the servers. These tools can help you identify business departments, which in turn can help you identify an owner.
- Related application owners:** Contact owners of applications that service a similar business department or function. Ask them to help you identify the roles that you need to fill. Even if you don't have an expert for a role in your organization, you must fill the role during the migration process. Business teams and IT teams should identify at least interim members, and then build a plan for ownership for long-term support of the workload after it migrates.

Scale roles for large migration initiatives

Depending on the size and number of workloads that you migrate, you might need to have multiple team members assigned to each role. A good approach is to use the scale that this article describes for up to five workloads that are medium in size and complexity per two-week sprint.

However, workload sizing and complexity can be challenging to judge. In your early migration waves, start with a core team, but scale out if needed.

If you find that you need to scale out, you should also plan for the roles that are described in the following table:

Role	Responsibilities
Program Manager	Organizes project management activities across multiple project scopes.
Migration Architecture Lead	Drives technical excellence across multiple migration architect scopes.

Responsibility matrix example

The following table uses this legend to indicate categories of responsibility per role for stages of a migration project:

- D = Driver:** One individual in the organization who is the single driver of the objective.
- A = Approver:** One or more individuals in the organization who make most decisions and who are responsible if the objective isn't met.
- C = Contributor:** Individuals in the organization who are responsible for carrying out tasks that support the objective.
- I = Informed:** Individuals in the organization who the project affects and who are regularly informed about decisions and the status of the project.

You can use the following responsibility matrix as a basis for your migration project. You might need to identify more roles or to shift responsibilities depending on the needs of your organization.

[Expand table](#)

Role	Digital estate discovery	Migration scope	Project plan	Migration tooling	Workload discovery	Workload assessment	Workload architecture	Wave planning	Workload test migration	Workload migration UAT	Workload migration	Workload release UAT
Migration Architect	D	D	A	D	A	A	D	A	A	A	A	A
Migration Engineer	C	I	C	C	D	D	C	D	D	C	D	C
Project Manager	I	I	D	I	I	I	I	I	I	D	I	D
Project Sponsor	A	A	A	A	I	I	A	I	I	I	A	I
User Acceptance Tester	I	I	I	I	I	I	I	I	I	C	I	C
Workload Architect	I	I	C	C	C	C	C	C	C	C	C	C
Workload Business Owner	I	I	C	I	A	A	A	A	A	A	A	A
Organizational Change Manager	I	I	C	I	I	I	I	I	I	C	I	C
Licensing Specialist	I	I	C	C	I	C	C	C	I	I	I	I
Cloud Operations Manager	C	C	C	I	I	I	I	C	I	I	I	I
Landing Zone Architect	I	I	C	C	I	I	C	C	I	I	I	I

Next step

Skills readiness for migration

Feedback

Was this page helpful?

Yes

No

Skills and support resources relevant to migration projects

Article • 04/10/2024

This article provides guidance on specific training resources to consider in preparation for your migration to Azure. It builds on the organizational skills that you learned during planning and preparing a migration journey. You might need to revisit the [planning migration articles](#) to make the most of the materials in this article. For more information on organizing your training plans, see [Build technical skills](#).

Self-led learning resources

Discover recommended learning resources to improve your migration skills.

Premigration skilling resources

The following tools can help your team complete assessment activities and plan for the overall adoption journey.

If you're still identifying what you need to migrate or modernize, these articles can help you understand the digital estate and the business justification for migration:

[] [Expand table](#)

Recommended article	Description
Balance the portfolio	Learn how to help ensure balance and proper investment allocations across an application portfolio.
Create a business case	Learn how to create and understand the business case that drives a cloud migration effort.
Rationalize the digital estate	Learn about cloud rationalization and the process of evaluating assets to determine the best approach for hosting assets in the cloud.

Migration skilling resources

The following learning resources can help prepare your team for migration activities:

Recommended resource	Description
Migrate to Azure	Learn how to migrate on-premises workloads to Azure.
The Migration Execution Guide	Find project resources that can help guide you through the decisions and steps of your migration.

Get technical support

- If you have questions or need help, [create a support request](#).
- If your support request requires deep technical guidance, see [Azure support plans](#) to align the best plan for your needs.

Get help from Microsoft and partners

Discover resources and support options for help with your migration.

Support option	Description
Azure cloud migration and modernization center	Get cloud migration tools and expert help to move, manage, and secure all your workloads, no matter the size of your business or industry.
FastTrack for Azure	Get direct assistance from Microsoft engineers, working hand in hand with partners to help customers build Azure solutions quickly and confidently. FastTrack brings best practices and tools from real customer experiences to guide customers from setup, configuration, and development to production of Azure solutions.
Cloud solution provider (CSP)	Use a CSP to take full advantage of the cloud. A CSP can assess your business goals for cloud adoption and identify the right cloud solution to meet your business needs, helping your business become more agile and efficient.
Azure expert partner	Get technical assistance, advice, and support for your migration project. Partners can also nominate you for Azure Migrate and Modernize and Azure Innovate offerings.

Next steps

[Assess workloads](#)

Feedback

Was this page helpful?

 Yes

 No

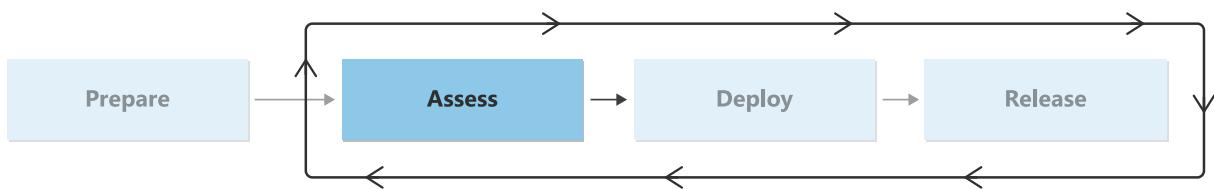
Migration assessment checklist

Article • 04/10/2024

Many of your existing workloads are ideal candidates for cloud migration. But not every asset is compatible with cloud platforms and not all workloads can benefit from hosting in the cloud. You can use [digital estate planning](#) to generate an overall migration backlog of potential workloads to migrate. However, this planning effort is high level. It relies on assumptions that the cloud strategy team makes and doesn't dig deeply into technical considerations.

Therefore, before you migrate a workload to the cloud, it's critical to assess the individual assets that are associated with that workload to determine whether they're suitable for migration.

In the assess phase, you evaluate the readiness of your workload and plan for the migrated state. After you complete this phase, you can deploy the workload for migration.



Your cloud adoption team should evaluate technical compatibility, the required architecture, performance and sizing expectations, and dependencies. Use this information to ensure that you can deploy the migrated workload to the cloud effectively.

If you don't complete these steps, migration activities during deployment might fail. You might deploy your workload in an insecure and unmanaged fashion, or your workload might not operate as intended due to missing services. You could also accrue unplanned costs. Therefore, it's vital to complete these activities before you begin to deploy resources to Azure.

Checklist

The following table provides an overview of the activities in the assess phase and the roles that are responsible for each activity.

[+] [Expand table](#)

Activity	Description	Responsible roles
Classify workloads	Instructions for classifying data sensitivity and mission criticality, and for identifying your organization's specific classifications	<ul style="list-style-type: none"> • Migration Architect • Cloud Operations Manager
Evaluate workload readiness	Instructions for assessment, common blockers, and common evaluation activities	<ul style="list-style-type: none"> • Migration Architect
Architect workloads	Guidance for designing the architecture of the migrated workloads, supporting services, and building and refining initial cloud estimates	<ul style="list-style-type: none"> • Migration Architect • Workload Architect • Landing Zone Architect • Cloud Operations Manager

Next step

[Classify workloads](#)

Feedback

Was this page helpful?

 Yes

 No

Classify workloads for a migration

Article • 04/10/2024

Each iteration of a migration process involves migrating one or more workloads. Before migration activities, it's important to classify each workload. Classification helps clarify governance, security, operations, and cloud-scale analytics requirements.

This article assumes that you have already [defined your tagging strategy](#).

During the migration assessment of a workload, you should verify the data sensitivity and mission criticality of each workload. You can capture this data in the tool that you use to track resources for migration. Ensure that resources are tagged appropriately during the migration.

These assessment data points can help other teams understand which workloads might require attention or support.

Data sensitivity

[Data classification](#) is based on how a data leak affects a business or customers.

Governance and security teams use data sensitivity or data classification as an indicator of security risks. During assessment, the cloud adoption team should evaluate the data classification for each workload that's targeted for migration and share that classification with supporting teams. Workloads that deal strictly in *public data* might not have any influence on supporting teams. As data becomes highly confidential, both governance and security teams likely have a vested interest in participating in the workload assessment.

Work with your security and governance teams as early as possible to define:

- A clear process for sharing any workloads in the backlog that have sensitive data.
- An understanding of the governance requirements and the security baseline that's required for different levels of data sensitivity.
- Any influence that data sensitivity might have on the subscription design, management group hierarchies, or landing zone requirements.
- Any requirements for testing data classification, which can include specific tooling or a defined scope of classification.

Mission criticality

[Workload criticality](#) is based on how significantly an outage affects a business. This data point helps operations management and security teams evaluate risks that involve outages and breaches. During assessment, the cloud adoption team should evaluate mission criticality for each workload that's targeted for migration and share that classification with supporting teams. *Low* or *unsupported* workloads are likely to have little influence on the supporting teams. However, as workloads approach *mission-critical* or *unit-critical* classifications, their operational dependencies become more visible.

Work with your security and operations teams as early as possible to define:

- A clear process for sharing any workloads in the backlog that have support requirements.
- An understanding of the operations management and resource consistency requirements for different criticality levels.
- Any influence that criticality might have on the subscription design, management group hierarchies, or landing zone requirements.
- Any requirements for documenting criticality, which might include specific traffic or usage reports, financial analyses, or other tools.

Data sovereignty

If your organization is subject to sovereignty and regulatory compliance, you need to classify your workloads and data based on your sovereignty requirements. For more information, see [Examples of data sovereignty requirements](#).

Organization-specific considerations

In addition to data and mission criticality, consider classifications that are specific to your organization. For example, many organizations with several development teams classify workloads by the project team who supports them, or by the line of business that they support.

Capture classifications

To apply the data classification labels to specific workloads in your migration tracking tools, see [What is data classification](#).

You should also determine the criticality of workloads by using the guidance for [business criticality in cloud management](#). You can use the operations management workbook or other tools for planning.

Next step

[Evaluate workload readiness](#)

Feedback

Was this page helpful?

 Yes

 No

Evaluate workload readiness

Article • 04/10/2024

This article focuses on how to evaluate the readiness of a workload to migrate to the cloud.

When you want to migrate a workload, the cloud adoption team ensures that all assets and associated dependencies are compatible with your deployment model and cloud provider. The team documents any required efforts to [remediate](#) compatibility problems.

Evaluation assumptions

Most of the content that discusses principles in the Cloud Adoption Framework for Azure is cloud agnostic. However, the readiness evaluation process must be specific to each cloud platform and to the migration tools that you selected in the [Prepare](#) phase.

The assessment tools that you selected should provide information about any blockers for migration. Common blockers include operating system support, server size, and data change rates that might affect replication.

Some organizations also face problems with configurations of virtual machines (VMs) that take advantage of the source hypervisor platform. These configurations include virtualization-based security, dynamic disks, non-Microsoft applications licenses, data source configurations, and certificates.

This article doesn't capture all possible evaluation activities because each environment and business outcome dictates specific requirements. To help you determine what those requirements are, here are a few common evaluation activities that are related to infrastructure, databases, and networks.

Evaluate cross-datacenter dependencies

If you're migrating workloads from multiple datacenters, you must assess any dependencies between those datacenters.

Consider the following capabilities to evaluate your cross-datacenter dependencies:

- **Visualize dependencies:** Use the [dependency visualization](#) capability in Azure Migrate and Modernize to pinpoint dependencies.

- **Group dependencies:** Use [dependency grouping](#) when you're dealing with global complexity. This capability helps you identify the IP addresses and ports of any assets that are required to support the workload.

 **Important**

- You need a subject matter expert who understands asset placement and IP address schemas to identify assets that reside in a secondary datacenter.
- You need to evaluate downstream dependencies and clients in the visualization to understand bidirectional dependencies.

Example scenarios

The following sections provide guidance for evaluating readiness to migrate workloads and databases to the cloud.

Common evaluation activities for Azure Migrate and Modernize

The following guidance assumes that you intend to migrate a workload to Azure. It also assumes that you're using Azure Migrate and Modernize for [replication activities](#).

You can use your Azure Migrate and Modernize project to assess workloads and calculate the cost of operating in Azure. For more information, see [Azure VM assessments in Azure Migrate and Modernize](#).

You can also use your Azure Migrate and Modernize project to assess readiness for migration, translate server size to Azure subscriptions based on actual use, and calculate costs. Further refine your cost calculations by [building a business case](#).

Be sure to document any discrepancies in host configuration, replicated VM configuration, storage requirements, or network configuration. Use that information to estimate the bandwidth considerations for your migration. Common components of bandwidth estimation include:

- **Total storage:** Calculate the total storage that replicated VMs need during the iterations that lead up to a release.
- **Drift or change rate:** Calculate the drift or change rate of storage that replicated VMs need during the iterations that lead up to a release.

- **Bandwidth requirements:** Calculate the bandwidth requirements that each iteration needs by summing the total storage and drift.
- **Unused bandwidth:** Calculate the available unused bandwidth on the current network to validate per-iteration alignment.
- **Migration velocity bandwidth:** Document the bandwidth that you need to reach the anticipated migration velocity. If you need any remediation to provide the necessary bandwidth, notify the team that's responsible for [remediation activities](#).

Note

Total storage directly affects bandwidth requirements during the initial replication. However, storage drift continues from the point of replication until release. This means that drift has a cumulative effect on available bandwidth.

For guidance on gauging bandwidth requirements, see [Common questions for migration and modernization tooling](#).

Common database evaluation activities

As part of your server migration, you might also look at migrating SQL Server instances or other database servers.

- **Document RPOs and RTOs:** Document the recovery point objectives (RPOs) and the recovery time objectives (RTOs) of the current database deployment. Use this information to help you make decisions during [architecture activities](#).
- **Document high-availability requirements:** Document high-availability configuration requirements. For more information about SQL Server requirements, see the [SQL Server high availability solutions guide](#).
- **Evaluate PaaS:** Evaluate platform as a service (PaaS) compatibility. The [Azure Database Migration Service guides](#) map on-premises databases to compatible Azure PaaS solutions, like Azure Cosmos DB, Azure SQL Database, Azure Database for MySQL, Azure Database for PostgreSQL, or Azure Database for MariaDB.
 - **PaaS compatibility without remediation:** When PaaS compatibility is an option without the need for any remediation, consult the team that's responsible for architecture activities. PaaS migrations can save time and reduce the total cost of ownership (TCO) of most cloud solutions.
 - **PaaS compatibility when remediation is required:** Consult the teams that are responsible for [architecture](#) and [remediation](#) activities. In many scenarios, the advantages of PaaS migrations for database solutions outweigh the increase in remediation time.

- **Document size and rate of change:** Document the size and rate of change for each database that you plan to migrate.
- **Document application and database dependencies:** When possible, document any applications or other assets that make calls to each database.

ⓘ Note

Synchronization of any asset consumes bandwidth during the replication processes. A common pitfall is to overlook how much bandwidth you need to keep assets synchronized between the points of replication and release. Databases are common consumers of bandwidth during release cycles, and databases with large storage footprints or a high rate of change are especially concerning.

Consider replicating the data structure with controlled updates before user acceptance testing (UAT) and release. In these scenarios, alternatives to Azure Site Recovery might be more appropriate. For more information, see [Azure Database Migration Service guides](#).

Next step

After you evaluate a system, the outputs feed the development of a new cloud architecture.

Architect workloads prior to migration

Feedback

Was this page helpful?

 Yes

 No

Design workload architecture before migration

Article • 04/10/2024

This article describes the process and considerations for designing the intended migrated state of a workload in the cloud. The article reviews activities that are part of defining a workload's architecture within an iteration.

The article about [incremental rationalization](#) indicates that some architectural assumptions are part of any business transformation that includes a migration. This article clarifies typical assumptions. It also points to a few roadblocks that you can avoid, and it identifies opportunities to accelerate business value by challenging architectural assumptions. This incremental model for designing architecture helps teams move faster and obtain business outcomes sooner.

Base architecture design on common assumptions

The following assumptions are typical for any migration effort:

- **Assume an infrastructure as a service (IaaS) workload.** Migrating workloads primarily involves moving servers from a physical datacenter to a cloud datacenter via an IaaS migration. The process typically requires minimal redevelopment or reconfiguration. This approach is known as a *lift and shift* migration.
- **Keep the architecture consistent.** Making changes to core architecture during a migration considerably increases complexity. Debugging a changed system on a new platform introduces many variables that can be difficult to isolate. Workloads should undergo only minor changes during migration, and any changes should be thoroughly tested.
- **Plan to resize assets.** Assume that few on-premises assets fully use any resource. Before or during migration, assets are resized to best fit actual usage requirements. Tools like Azure Migrate and Modernize provide sizing based on actual use.
- **Capture business continuity and disaster recovery (BCDR) requirements.** If you have an agreed-on service-level agreement (SLA) for the workload already negotiated with the business, continue to use the SLA after the migration to Azure. If an SLA isn't already set, define one before you design the workload in the cloud to make sure that you design appropriately.
- **Plan for migration downtime.** Like failing to meet SLA requirements, downtime that occurs when you promote a workload to production can have an adverse

effect on the business. Sometimes solutions that must transition with minimal downtime need architecture changes. Before you begin release planning, assume that a general understanding of downtime requirements is established.

- **Define user and application traffic patterns.** Existing solutions might depend on existing network routing patterns and solutions. Identify the resources that you need to support current network usage.
- **Plan to avoid network conflicts.** When you consolidate datacenters into a single cloud provider, you're likely to create conflicts in Domain Name System (DNS) or other networking structures. During migration, it's important to design to avoid these conflicts to avoid interruptions to production systems that are hosted in the cloud.
- **Plan for routing tables.** Make sure that your project includes a plan for modifying routing tables if you consolidate networks or datacenters. Consider cross-datacenter routing requirements.

Design architecture for a landing zone

In the [Ready phase](#) of the Cloud Adoption Framework, your organization deployed shared platform services as part of adopting [Azure landing zones](#). In [Ready your landing zone for migration](#), you prepared the landing zone to receive migrated workloads.

Based on this planning, you can assume that the following migration components are in place:

- Hybrid connectivity connects your Azure networks to your on-premises networks.
- Network security appliances like Azure Firewall handle the inspection of traffic outside your workload.
- Azure policies to enforce governance practices like logging requirements, allowed regions, disallowed services, and other controls are active.
- An Azure Monitor Logs workspace for shared logging is set up in Azure Monitor.
- Shared identity resources to support domain-joined servers or other identity needs are established.

If these migration essentials aren't established, your organization should immediately revisit the Ready phase to address these needs. Without these components, your migration likely will have delays and setbacks and be less successful.

The workload that you're designing has a subscription assigned to it, ideally through a [subscription vending process](#). The subscription might contain several workloads or just one workload depending on how your organization completed its [resource organization for workloads](#). If you migrate a workload that has many application environments, you

might even have multiple subscriptions based on the guidance for [application environments](#).

Design workload network architecture

Plan to deploy application-specific resources to a specific subscription, and plan to design a dedicated virtual network for the workload. For more information, see guidance for [designing your networking architecture](#). You should especially focus on [segmenting virtual networks](#).

Your network likely needs resources like load balancers and other application-delivery resources. You can use the [N-tier architecture guide](#) to plan these resources in Azure.

Design workload dependencies

Your migration assessment tools should give you a way to do dependency analysis, like [dependency analysis](#) in Azure Migrate and Modernize. The tool helps you identify interdependencies between servers.

In addition to planning architecture for the workload itself, you might need to consider workload-to-workload dependencies. For example, some workloads might need frequent communication. If so, planning your migration waves and dependencies to account for this requirement helps make your migration successful.

You can use guidance in Azure Architecture Center, such as for [spoke-to-spoke](#) networking, to design how interconnected workloads work after migration.

Prepare for adopting confidential computing

A subset of your workloads with sovereignty requirements might lead to using confidential computing. As part of a sovereign landing zone deployment, management groups for confidential computing are created.

Within a sovereignty context, using confidential computing requires Azure Key Vault and customer managed keys as a supporting service. For more information, see [encryption with customer-managed keys in Microsoft Cloud for Sovereignty](#).

Update your initial cloud estimate

As you finish your architecture design, revisit your cloud estimate to make sure that you're still within the planned budget. As you add supporting services like load

balancers or backups, costs can change. Although you can use tools like [business cases](#) in Azure Migrate and Modernize to do detailed cost management exercises, you should periodically revisit your estimates as you adjust your architecture design.

If you're familiar with traditional IT procurement processes, estimating resources in the cloud might seem foreign. When you adopt cloud technologies, acquisition shifts from a rigid, structured capital expense model to a fluid operating expense model. Planning a migration to the cloud often is the first time an organization or IT team encounters this change.

In the traditional capital expense model, an IT team attempts to combine buying power for multiple workloads across various programs. This approach centralizes a pool of shared IT assets that can support each of those solutions. In the operating expense cloud model, costs can be directly attributed to the support needs of individual workloads, teams, or business units. It gives an organization a more direct attribution of costs to internal customers and the business objectives that they support. This more dynamic approach to financial management is often called *financial operations (FinOps)*. Although FinOps isn't an Azure-specific consideration, it can be helpful to have an expanded understanding of FinOps. For more information, see [What is FinOps?](#).

When you design your workload architecture for migration, you can use the [pricing calculator](#) with your assessment information to understand the price of the entire workload.

Also, after your workload is migrated, you should continue to work to optimize workload costs. For more information about how organizations can mature their cost management skills, see [Improve the cost management discipline](#).

Know when to change your architecture

Migrations tend to focus on maintaining an existing architecture and transitioning it to your cloud platform. However, there are times when you might need to rearchitect your workload, even for migration. This list gives examples of when you might need to make architectural changes before you migrate:

- **Paying for technical debt.** Some aging workloads carry a high amount of technical debt. Technical debt can lead to long-term challenges by increasing hosting costs with any cloud provider. When technical debt unnaturally increases hosting costs, you should evaluate alternative architectures.
- **Improving reliability.** Standard operation baselines provide a degree of reliability and recovery in the cloud. Some workload teams might require higher SLAs, which could lead to architectural changes.

- **High-cost workloads.** During migration, you should optimize all assets to align sizing with actual usage. Some workloads might require architectural modifications to address specific cost concerns.
- **Performance requirements.** When workload performance directly affects business, extra architectural consideration might be required.
- **Secure applications.** Security requirements tend to be implemented centrally and are typically applied to all workloads in a portfolio. Some workloads might have specific security requirements that can lead to architectural changes.

Each of these criteria serves as an indicator of a potential migration roadblock. In most cases, you can address the criterion after you migrate the workload by right-sizing servers, adding new servers, or making other changes. However, if any of those criteria are required before you migrate a workload, remove the workload from the migration wave and evaluate it individually.

The [Azure Well-Architected Framework](#) and [Azure Well-Architected Review](#) can help guide conversations with the technical owner of a specific workload to help them consider alternative options for deploying the workload.

The workload is then classified as a rearchitecture or modernization effort in your cloud adoption plan. Because of the extra time it takes to rearchitect a workload, consider these alternative workload adoption paths as separate from the migration process.

Architecture checklist

You can use the following checklist to make sure that you cover critical architectural considerations:

- Confirm that your architecture meets SLAs for availability, disaster recovery, and data recovery.
- Confirm that you applied network segmentation practices to your network design.
- Confirm that you planned for monitoring and log capturing.
- Confirm that your virtual machines are sized appropriately for the actual computing time that you need.
- Confirm that your disks are sized appropriately for the actual size and performance that you need.
- Confirm that you planned for load balancing services if they're needed. The services might include instances of Azure Load Balancer, Azure Application Gateway, Azure Front Door, or Azure Traffic Manager.
- Confirm that you planned for sovereignty and confidential computing requirements if they're needed.

Next step

[Deploy migration workloads](#)

Feedback

Was this page helpful?

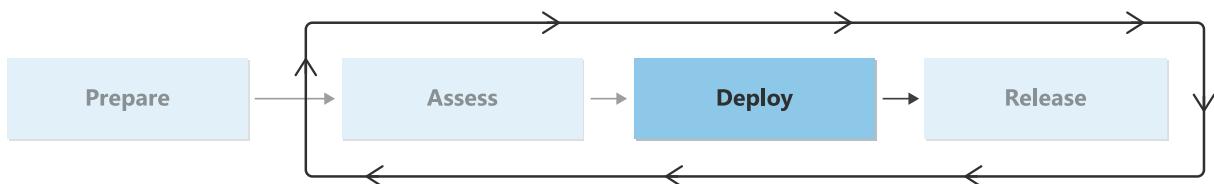
 Yes

 No

Migration deployment checklist

Article • 04/10/2024

In the deploy phase of your adoption journey, you begin the technical implementation of the migration. You can use the workload architecture and assessment materials from the assess phase to begin deployment and remediation of problems. You can also begin to replicate servers to Azure, prepare them for migration, and run your technical tests.



This phase can be the most technically demanding. It's important to make sure you complete your remediation activities and do detailed testing. You should have a clear understanding of how your services operate together. By the end of this phase, you should have high confidence that your workload is ready to operate in Azure and that you're ready to plan for the workload release.

Checklist

The following table provides an overview of the activities in this phase and the roles that are responsible for each activity.

[Expand table](#)

Activity	Description	Responsible roles
Deploy supporting services	Guidance about how to plan for and deploy non-server resources for your workload	<ul style="list-style-type: none">• Migration Engineer
Remediate assets	Guidance to help build your remediation plan based on assets	<ul style="list-style-type: none">• Migration Architect• Migration Engineer• Project Manager
Replicate assets	Guidance for replicating source servers to the cloud	<ul style="list-style-type: none">• Migration Engineer
Prepare for management	Guidance on planning for management as part of the migration	<ul style="list-style-type: none">• Cloud Operations Manager• Migration Engineer

Activity	Description	Responsible roles
		<ul style="list-style-type: none">• Workload Architect
Test your migration	Guidance on how to do technical migration testing before doing business testing	<ul style="list-style-type: none">• Project Manager• Workload Architect• Migration Engineer

Next step

[Deploy supporting services](#)

Feedback

Was this page helpful?

 Yes

 No

Deploy supporting services

Article • 04/10/2024

Every workload you migrate to the cloud requires several supporting services.

You must place replicated or staged servers into virtual networks, and many workloads require load balancers and an application delivery solution. Also, there are considerations for how to [prepare to manage](#) workloads.

When you designed your [workload architecture](#), you identified these services. Before you begin to migrate resources, *deploy* the services to ensure that you have everything ready for your workload to operate successfully.

This article assumes that you prepared your shared platform solutions by [readying your landing zones for migrations](#) and completed your workload architecture.

Plan for supporting service deployment

It's sometimes unclear when to deploy the supporting services. Some services, like Azure Virtual Network, are based on consumption. Other services, like Azure Application Gateway, are based on runtime, even if you aren't actively using them. If they're powered on, they incur cost.

So, you should plan for the following situations:

- **Services to deploy early in the process**
 - Core, no-cost, and consumption-based services, like virtual networks, network security groups, and storage accounts.
 - Services that can be paused or put in standby mode. Pause them until ready.
- **Deprovision after use**
 - Services that begin billing immediately for testing purposes. Deprovision them when not needed. Use infrastructure as code (IaC) to consistently deploy these resources.

Required services

While your workload architecture provides a definitive list of services, you can use the following options as a guide to help identify your supporting services:

- **Resource organization:**
 - A subscription to hold your workload

- Various resource groups to organize your resources
- **Networking:**
 - A virtual network with subnets that match your network design
 - Network security groups assigned to the subnets and restricted to required traffic only
 - A peering connection to your hub virtual network
 - Any necessary user-defined routes to shape traffic, such as to send your default route traffic to a firewall appliance
 - Any application presentation resources for web applications, such as Application Gateway or Azure Front Door
 - Load balancers for workloads that have multiple nodes
- **Identity and security:**
 - Any key vaults to hold certificates or secrets
- **Management:**
 - Any workload-specific Azure Monitor Logs workspaces
 - Any workload-specific Azure Site Recovery and Azure Backup instances
 - Any workload-specific Azure Update Manager instances
 - Any workload-specific alerts

Also, review [Prepare for management](#) to understand the management-specific considerations.

Next steps

[Remediate assets](#)

Feedback

Was this page helpful?

 Yes

 No

Remediate assets prior to migration

Article • 04/10/2024

During the migration assessment process, the team identifies any configurations that might make an asset incompatible with the chosen cloud provider. Remediation is a checkpoint in the migration process that you can use to resolve any incompatibilities.

This article describes a few common remediation tasks and helps you decide if remediation is a wise investment.

Remediation types

There are two main types of remediation activities that you need to plan for throughout your deployment.

- **Based on the results of the assessment activities**
 - Remediation activities that need to be completed to allow replication and deployment.
 - You determined these remediation activities in your workload assessment during the assess phase. You must perform these tasks to ensure that you can properly replicate and stage your workload in the cloud.
 - This is focused primarily on the source servers being migrated.
- **Based on the results of the testing activities**
 - This comes from [testing migration activities](#) and performing [business testing](#).
 - These remediation activities are focused on the configuration of the replicated destination servers and any assisting services like load balancers, virtual networks, and storage accounts.
 - These tasks are likely more iterative. Testing and remediating through several cycles until all test cases pass is expected.

Track remediation activities

Throughout the iteration, you can identify remediation tasks for your workloads through assessment or testing. You need to track these tasks as project activities to make sure that they're completed.

While small migration waves can use spreadsheets to track items, larger waves with many remediation tasks generate multiple items. You can use tools like [Azure DevOps](#) to create and prioritize work items and move through specific phases to help you scale out.

Even if you don't use Azure DevOps for other efforts, you can use it to triage remediation problems and organize tasks for the migration process.

As you create these tasks, you should make sure to connect them back to the workload that they affect. This allows you to assess which workloads might be delayed by remediation tasks. You can then prioritize the work by workload priority.

Some problems might affect multiple workloads. These are generally items with the host, a broad configuration, or problems with the landing zone as a whole. These problems should be the first ones prioritized for remediation.

Common remediation tasks

Technical debt is a healthy and expected part of the corporate environment. Architecture decisions that are suited for an on-premises environment might not be suitable in a cloud platform. In either case, common remediation tasks might be required to prepare assets for migration. The following are a few examples:

- **Minor host upgrades:** Occasionally an outdated host needs to be upgraded prior to replication.
- **Minor guest operating system upgrades:** You probably need to patch or upgrade your operating system before replication.
- **Service-level agreement (SLA) modifications:** Backup and recovery processes change significantly in a cloud platform. The backup processes for migrated assets might need to be modified to ensure that they continue to achieve their necessary SLAs in the cloud.
- **Application configuration changes:** Migrated applications might require adjustments to variables such as network paths to dependent assets, service account changes, or updates to dependent IP addresses.
- **Minor changes to network paths:** Routing patterns need to be modified to properly route user traffic to the new assets. This isn't production routing to the new assets, but configuration to allow for proper routing to the assets in general.

Large-scale remediation tasks

There's little need for remediation when a datacenter is properly maintained, patched, and updated. Remediation-rich environments tend to be common within large enterprises. This can include organizations under large IT downsizing, legacy-managed service, and acquisition-rich environments. In each of these environments, remediation comprises a large portion of the migration effort. The following remediation tasks might frequently occur or negatively affect migration speed or consistency. If this happens,

separate remediation into a parallel effort and team similar to cloud adoption and cloud governance.

- **Frequent host upgrades:** Upgrading multiple hosts to complete a workload's migration can delay the migration team. Isolate affected applications and address the remediation steps before you include affected applications in any planned releases.
- **Frequent guest operating system upgrade:** Large enterprises commonly have servers running on outdated versions of Linux or Windows. Aside from the security risks of operating an outdated operating system, there are also incompatibility problems that prevent the migration of affected workloads. When multiple virtual machines (VMs) require operating system remediation, try separating these efforts into a parallel iteration. Some upgrades can be completed by the migration tooling as part of the migration process, such as the [Windows Server upgrade](#) feature in Azure Migrate and Modernize.

Address large-scale remediations

Because remediation for smaller workloads can be straightforward, choose smaller workloads for your initial migration waves. As your migration efforts mature and you begin to tackle larger workloads, remediation can be time consuming and costly. For example, remediation efforts for a Windows Server 2003 migration involving a pool of assets with more than 5,000 VMs can delay a migration by months. When such large-scale remediation is required, you might need to change your plans for migrating the affected workloads. In such instances, modernization activities to maximize the value of remediation efforts might be more efficient and productive.

You can use the following questions to help guide decisions:

- Have all workloads affected by the remediation been identified and notated in the migration backlog?
- For workloads that aren't affected, does a migration produce a similar return on investment (ROI)?
- Can the affected assets be remediated in alignment with the original migration timeline? What effect do timeline changes have on the ROI?
- Is it economically feasible to remediate the assets in parallel with migration efforts?

If the previous questions aren't answered, consider these modernization approaches:

- **Containerization:** Some assets can be hosted in a containerized environment without remediation. This might produce less-than-favorable performance and doesn't resolve security or compliance issues.

- **Automation:** Depending on the workload and remediation requirements, scripting the deployment to new assets using a DevOps approach might be more profitable.
- **Rebuilding:** When remediation costs and business value are equally high, a workload is a good fit for rebuilding or rearchitecting.

Next step

[Replicate assets](#)

Feedback

Was this page helpful?

 Yes

 No

Replicate assets in a cloud migration

Article • 04/10/2024

On-premises datacenters store physical assets, like servers, appliances, and network devices. But each physical asset, like a server, is only a shell. The real value comes from the binary that runs on the server. The datacenter exists because of the applications and data, which are the primary binaries that you migrate. Digital assets and binary sources, like operating systems, network routes, files, and security protocols, power the applications and data stores.

The replication process consists of the following steps:

- 1. Replication:** Copies a point-in-time version of various binaries.
- 2. Seeding:** Copies the binary snapshots to a new platform and deploys them onto new hardware. The seeded copy of the binary behaves exactly like the original binary on the old hardware. But the snapshot of the binary is out of date and misaligned with the original source.
- 3. Synchronization:** Aligns the new binary and the old binary. This process continuously updates the copy that's stored on the new platform. Synchronization continues until the asset is promoted in alignment with the chosen promotion model. At that point, the synchronization stops.

Prerequisites for replication

Before replication, you should complete the activities in the [prepare](#) and [assessment](#) phases. To begin replicating, you need to have:

- A subscription for your migrated resources.
- A migration tool to move the binary copies over.
- The source binaries, prepared for replication and synchronization. Their exact configuration depends on your migration tool. Preparation includes remediating any replication problems that you found in the assessment phase. For an example of how to initiate replication, see [Migrate from VMware via agentless migration](#).
- Any dependencies for your workload that you identified during the [workload architecture design](#) step. These dependencies can include resource groups, virtual networks, and subnets in which you intend to deploy the replicated virtual machines. For more information, see [Deploy supporting services](#).

Replication risks: Physics of replication

When you plan and perform binary source replication to a new destination, consider the following fundamental laws:

- **Speed of light:** When you move high volumes of data, fiber is the fastest option. But fiber cables can only move data at two-thirds the speed of light. There's no method for instantaneous or unlimited replication of data.
- **Speed of the WAN pipeline:** The uplink bandwidth is even more important than the speed of data movement. The volume of data per second that your company's existing WAN transfers to the target datacenter determines the uplink bandwidth.
- **Speed of WAN expansion:** If budget allows, you can add more bandwidth to your company's WAN solution. But it can take weeks or months to procure, prepare, and integrate more fiber connections.
- **Speed of disks:** Even with infinite data speed and an infinite bandwidth limit between the source binary and the target destination, physics still limits replication. Data replication occurs only as quickly as source disks can read the data.
- **Speed of human calculations:** Disks and light move faster than human decision processes. When a group of people collaborate and make decisions together, the results come slowly. Replication can't overcome delays related to human calculations.

Each of these laws of physics drives the following risks that commonly affect migration plans:

- **Replication time:** Replication requires time and bandwidth. Plans should include realistic timelines that reflect the amount of time it takes to replicate binaries.

Total available migration bandwidth is the amount of up-bound bandwidth that other higher priority business needs don't consume. The up-bound bandwidth is measured in megabits per second (Mbps) or gigabits per second (Gbps). *Total migration storage* is the total disk space, measured in gigabytes (GBs) or terabytes (TBs), required to store a snapshot of assets to be migrated.

To determine an initial time estimate, divide the *total migration storage* by the *total available migration bandwidth*. Note the conversion from bits to bytes. The next item describes a more accurate calculation of time.

- **Cumulative effect of disk drift:** From the point of replication to the promotion of an asset to production, the source and destination binaries must remain synchronized.

Drift in binaries consumes extra bandwidth because you must replicate changes to the binary on a recurring basis. During synchronization, the calculation for total migration storage includes all binary drift. The longer it takes to promote an asset to production, the more cumulative drift occurs. The more synchronized assets that you have, the more bandwidth you consume. For each asset in a synchronization state, you have less total available migration bandwidth available.

- **Time-to-business change:** Synchronization time has a cumulative negative effect on migration speed. Prioritization of the migration backlog and advanced preparation for the [change communication](#) plan are crucial to the speed of migration.

The most significant test of business and technical alignment during a migration effort is the pace of promotion. The faster an asset is promoted to production, the less disk drift affects bandwidth. And you can allocate more bandwidth and time to the next workload's replication.

Plan for when data requirements exceed network capacity

In a cloud migration, you replicate and synchronize assets over a network between an existing datacenter and the cloud. The existing data size requirements of various workloads might exceed network capacity. In such a scenario, the migration process might be radically slowed, or in some cases, stopped entirely.

If your assessment, initial replication, or testing identifies a capacity problem, consider using [Azure Data Box](#) to transfer independent data stores. Use this approach to ship large volumes of data to the cloud before the workload migration.

Some non-Microsoft partner solutions also use Data Box for migrations. With these solutions, you can move a large volume of data via an offline transfer, but you synchronize it later at a lower scale over the network.

Next step

[Prepare for management](#)

Feedback

Was this page helpful?

 Yes

 No

Prepare for management activities

Article • 04/10/2024

You should prepare to carry out management activities after a workload migration is complete. Organizations that fail to plan for management can experience replicated or delayed workloads. Furthermore, when resources are migrated but not managed, problems such as outages, breaches, or performance problems can occur.

Note

The guidance in this article builds on the broader Cloud Adoption Framework for Azure management guidance. These additional guides might be relevant to building your management routines:

- [Azure landing zones: Management design area](#)
- [Azure Management Guide](#)
- [Cloud Adoption Framework management methodology](#)

Ensure you have a management routine in place for each workload.

Minimum management goals

For each workload, these are the minimum management goals you should plan and prepare for:

- **Logs:** Appropriate logs are being collected by your logging system, like Azure Monitor Logs. Logs should include:
 - Activity logs to observe changes and events on services.
 - Diagnostic logs to collect metric and service-specific logs.
 - System logs from migrated or newly created virtual machines.
- **Alerts deployment:** Deployment of general alerts based on organizational standards and alerts for your workload.
- **Backup:** Backup configurations and processes that you can use to restore state in compliance with your organization's service-level agreements (SLAs).
- **Business continuity and disaster recovery (BCDR):** A BCDR configuration that allows you to fail over and restore service in compliance with your organization's SLAs.
- **Security posture:** Enrollment in the Microsoft Defender for Cloud policies selected by your organization to perform security posture management and vulnerability

detection. This might include the deployment of vulnerability detection services or the Microsoft Defender for Endpoint agent.

- **Serial console for Azure Virtual Machines:** Enabling a [serial console](#) in Virtual Machines to help with troubleshooting.
- **Enable automatic shutdown:** Configuration of automatic shutdown schedules for virtual machines that meet business requirements.
- **Tags:** Deployment of tags and remediation of incorrect tags.
- **Update management:** The periodic update of virtual machines such as with Azure Update Manager.

If you identify a missing management goal, you should add it to your remediation plan for the workload.

Organize transition to management

To organize a smooth transition to management for your migrated workloads, you should have a plan in place to ensure that you:

- **Involve application and workload owners:** The application owners are aware of any changes to activities to the servers being migrated such as deployment of code or installation of modules.
- **Involve monitoring, management, and security teams:** Teams responsible for monitoring, management, and security are aware of changes and responsibilities after the migration.
- **Define and communicate roles and responsibilities:** All teams understand their responsibilities during the migration.

Next step

[Migration testing](#)

Feedback

Was this page helpful?

 Yes

 No

Test your migration deployment in Azure

Article • 12/03/2024

After you replicate or stage your workloads and ensure that supporting services are available, you can begin your migration testing. Migration testing primarily focuses on two areas:

- **Architecture:** Test your architecture to ensure that it works with the replicated or staged resources.
- **Management routines:** Test your management plan for the migrated resources to ensure that it's functional.

Unlike [business testing](#), migration testing focuses on IT activities.

As you identify problems, you can add them to your [remediation plan](#). After you address all the problems, you can proceed to the workload release.

Perform test migrations

After you replicate resources, you can perform test migrations in isolated environments to ensure that you don't affect production workloads.

Test migrations vary depending on the tooling, but generally you create a replica of your source systems that runs in parallel to the live systems. Perform tests on these secondary systems. When you complete testing, you can clean up the replicated resources without introducing any permanent changes.

To do tests, you need:

- **An isolated network** where you test failover. Match the network configuration to the intended migration network configuration as much as possible.
- **Isolated network access** from a source, like a point-to-site VPN, a jump box, or Azure Bastion.
- **An authentication mechanism** to authenticate to the test environment. The test environment is isolated, so it can't use your landing zone's identity provider.

You might use a test-migrated domain controller that you deploy to the test environment with the test migration resources. After testing, clean up the domain controller with the resources.

Alternatively, your isolated network might have a test domain controller in it. Peer the network to allow for replication of Active Directory traffic. You can take a snapshot of the domain controller in Azure, and then delete the peer for testing purposes to isolate the network. You can seize any necessary roles, and then restore the state when you complete testing to avoid making changes to the live identity provider.

Your migration tool should be able to perform a test migration, and clean up the test materials. For an example of such a test migration process that would work in Azure Migrate, see the [Test migrations for VMware agentless migrations](#). This gives a starting point for understanding how the tools can help you with test migrations.

💡 Tip

You can also use this testing environment for [business testing](#).

Remediate testing problems

After you do testing, make sure that you:

- **Record any discovered problems** in the remediation plan.
- **Triage problems** based on their severity, and identify any workarounds as part of the triaging.
- **Document workarounds**. If you can incorporate the workaround as part of the migration, you might not need to remediate the problem.
- **Start with non-workaround items**. Consider remediating items without workarounds first.

Example testing plan

Here's a basic example of a testing plan output for a migration project:

 Expand table

Test	Successful/unsuccessful	Note
Virtual machines deploy	<input checked="" type="checkbox"/>	
Administrators can sign in to virtual machines	<input checked="" type="checkbox"/>	

Test	Successful/unsuccessful	Note
Internet Information Services (IIS) web services start	✓	
Service 1 starts	✓	
Service 2 starts	✗	Service had to be manually started
Website access	✓	
SQL services start	✓	
Database access	✓	
Load balancing between websites works	✓	
Ingress from Azure Application Gateway works	✗	Application Gateway has a certificate problem
Total time for the test transaction was less than 5 ms	✓	

Next step

[Release migrated workloads](#)

Feedback

Was this page helpful?

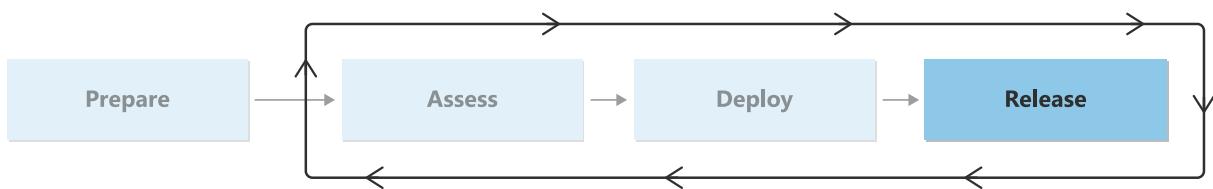
 Yes

 No

Migration release checklist

Article • 04/10/2024

This phase guides you through releasing your deployed workloads to production use. It includes the technical processes of taking your final migration steps, communicating to the business, making final change approvals, cleaning up resources, and learning from the migration.



Conduct the migration

When you reach this phase, you have your resources deployed into Azure, either as new resources with replicated data in a properly configured state or as a replicated server through a migration tool. From a technical perspective, all that remains to do is to start the final migration tasks and send traffic over to the new location.

However, you need to do more to make sure the migration is successful. Clear communication with the business and support teams is critical, as is finalizing activities, including a proper retrospective, after the migration.

Failure to communicate, test, or successfully migrate resources can create outages and business disruptions. A failure to decommission retired assets or optimize workloads can lead to a failure to achieve the business outcomes of the migration. To avoid these problems, you need to finish the migration strong.

Checklist

The following table gives an overview of the activities that are documented in this phase and the roles that are responsible for carrying out each activity.

[\[\] Expand table](#)

Activity	Description	Responsible roles
Plan change communication	Contains guidance and examples for how to communicate the upcoming change to the business and IT teams.	<ul style="list-style-type: none">• Project Manager• Organizational Change Manager

Activity	Description	Responsible roles
Conduct business testing	Contains guidance on how to conduct business testing to ensure that the functionality and usability of the workload remains the same after you migrate it to the cloud.	<ul style="list-style-type: none"> • Project Manager • Workload Business Owner • Workload Architect • User Acceptance Tester
Complete the migration	Contains guidance for how to conduct your migration release window.	<ul style="list-style-type: none"> • Migration Architect • Migration Engineer
Optimize costs after the migration	Contains resources for optimizing workload costs after the migration.	<ul style="list-style-type: none"> • Workload Architect • Cloud Operations Manager • Project Manager
Conduct a retrospective	Contains guidance on how to do retrospectives to improve the migration of future workloads.	<ul style="list-style-type: none"> • Workload Architect • Cloud Operations Manager

Next step

Now that you have a general understanding of the optimization process, you're ready to begin communicating the changes.

[Change communication](#)

Feedback

Was this page helpful?

 Yes

 No

Change communication

Article • 04/10/2024

Change communication means informing the business about your upcoming changes to ensure that everyone affected by the transition understands the cloud migration process.

Your communication plan can remain consistent, but each workload should undergo its own change communication because of its individual users and operators.

The change communication is in addition to any change requests captured by your change management team. You should work with your change management team to ensure that they properly manage the change. That team might also be able to help with the change communication.

Audience for change communication

Your change communication needs to target various groups in your organization. You should have a clear plan for each of the following roles:

- **Business users** who use the workload in question to complete activities
- **Change managers** who need to plan for risk management and must ensure that the cutover follows all necessary requirements
- **Owners and operators** of the assets moving to Azure
- **Developers** who deploy to the assets moving to Azure
- **Migration engineers** who perform the migration

ⓘ Note

Your organization might also have other necessary roles to include.

Change communication content

In your change communication, you should make it clear to each role how the upcoming change specifically affects them and what they must do to prepare. Your change communication should also include groups that work with related systems or in related business areas. Even if they don't have preparation steps and won't be directly affected by the change, they should be aware in case there are any indirect impacts to their work.

You should answer the following questions in your change communication:

- **Critical dates:** What are the critical dates for the migration?
- **Impact:** Whose work is disrupted, when, and for how long?
- **Steps to completion:** What work should each role complete before the change in order to be ready?
- **Post-change steps:** What work should each role complete after the change to confirm functionality?
- **Get support or learn more:** Who should individuals reach out to if they have questions or challenges?

Due to the number of roles involved, you might need to send out different change notifications to different audiences. There might be a broad communication for a general audience that explains the time frame and expected outages, and then a more specific communication for IT resources to plan for operational actions.

Example change communication content

You can use the following template to draft your change communication:

 Expand table

What to communicate	Details
What is changing?	Application X is migrating to Azure!
When is this change happening?	February 29, 2024, at 10 PM PST
Who will this change affect?	Business users of Application X and anyone who uses systems that pull information from Application X.
How will they be affected?	Application X won't be available for new work until systems are restored on March 1, 2024, at 2 AM PST.
What do I need to do?	Make sure your work in Application X is saved before the change. You can resume normal functions on March 1. Reach out if you encounter any problems.
Who should I reach out to?	Contact < <i>email</i> > for questions or more information before the change. If you encounter problems after the change, contact Contoso Helpdesk for assistance. Mention Application X in the subject line.

Next step

Business testing, or user acceptance testing (UAT), can begin after you document and plan the business change.

Business testing

Was this page helpful?

 Yes

 No

Perform business testing during a migration

Article • 04/10/2024

While the migration team facilitates the migration of a workload to Azure, testing of a workload is best performed by the business users of that workload. The migration team supports this business activity by facilitating workload testing, developing testing plans, and automating tests when possible.

During business testing, or user acceptance testing (UAT), you observe real users attempting to use the new solution in the context of a real or replicated business process.

ⓘ Note

Automated testing isn't always available.

Automated testing is an efficient way to test any system. However, cloud migrations often focus on legacy systems or stable production systems. These systems are seldom managed by thorough and well-maintained automated tests.

This article assumes that automated tests are available at the time of migration.

The goal of business testing is to obtain validation from power users to certify that the new solution performs in line with expectations and doesn't impede business processes. If that goal isn't met, the business testing serves as a feedback loop that can help you define how and why the workload isn't meeting expectations.

Business activities during business testing

During business testing, the first iteration is manually driven directly with customers. This is the purest and most time-consuming form of feedback loop.

- **Identify power users:** The business generally has a better understanding of the power users who are most affected by a technical change. Power users are the people who frequently perform a real-world process that requires interactions with a technology tool or set of tools, such as a call center that services customers.
- **Align and prepare power users:** Ensure that power users understand the business objectives, desired outcomes, and expected changes to business processes. Prepare power users and their management structure for the testing process.

- **Engage in feedback loop interpretation:** Help the IT staff understand the effects of various points of feedback from power users.
- **Clarify process change:** Communicate the process change and any downstream effects when transformation could trigger a change to business processes.
- **Prioritize feedback:** Help the IT team prioritize feedback based on the business impact.

Migration team activities during business testing

The migration team is one of the recipients of the business testing output. The feedback loops exposed during business testing eventually become work items that define technical change or process change. As a recipient of the business testing output, the migration team is expected to aid in facilitation, feedback collection, and management of resultant technical actions.

Typical activities that the migration team performs during business testing include:

- **Provide structure and logistics:** After working with power users, you can create a spreadsheet with places for testers to add their notes that define the tests that need to be run.
- **Facilitation:** Aid in facilitation during testing.
- **Record feedback:** Provide a means and process for recording feedback. You can use Azure DevOps or an Excel spreadsheet to effectively capture information and record action items.
- **Prioritize feedback:** Help the business prioritize and validate feedback.
- **Plan for changes:** Develop plans for acting on technical changes.
- **Identify automated tests:** Identify existing automated tests that might streamline the testing by power users.
- **Process improvement:** Study testing processes, define benchmarks, and create automation to further streamline power user testing for changes that might require repeated deployment or testing.

Example testing plan

You can use the following as a reference to build your testing plan:

[] Expand table

Test	Steps	Successful/unsuccessful	Tester notes
Sign in	Sign in using your normal credentials. Confirm that you see the same information as the production system.	✓	No remarks
Create new record	Create a new record in the system and confirm that all workflow jobs run appropriately.	✓	No remarks
Read created record	Reopen the record you created and confirm that all materials are filled in.	✓	No remarks
Update created record	Make changes to the record you created and confirm that the update proceeds accurately.	✓	No remarks
Delete created record	Delete the record you created and confirm that the record is removed from the system.	✓	No remarks
Read existing record	Open one of the existing test records that start with "Test-Record-Azure-Test" and confirm you can read all materials.	✓	No remarks
Update existing record	Update the test record that you used in the previous step and confirm that the update proceeds accurately.	✗	Update failed with error message: "Unable to access transaction database."
Delete existing record	Delete the test record that you used in the previous step.	✗	Update failed with error message: "Unable to access transaction database."

Next step

[Complete migration](#)

Feedback

Was this page helpful?

 Yes

 No

Complete the migration

Article • 04/10/2024

Promotion or cutover to production marks the completion of a workload's migration to the cloud. After you promote the asset and all of its dependencies, production traffic is rerouted. After the traffic is rerouted, the on-premises assets become obsolete, so you can decommission them.

The promotion process varies according to the workload's architecture. However, you can expect several consistent prerequisites and a few common tasks. This article describes each prerequisite and task and serves as a migration checklist.

Make sure you add workload-specific considerations to your checklist.

Migration window playbook

- **Send pre-promotion communications.** Although you already communicated about the change window, make sure to send a notification to all necessary parties so that they know the promotion has begun.
- **Validate the resources.** Confirm that all staged resources are in a functioning state. These resources include storage accounts and network security groups.
- **Pause monitoring.** While you migrate the workload, you'll likely create a temporary outage. Pause monitoring to prevent noise.
- **Take final replication steps.** Depending on your promotion method, you might need to take final replication steps to handle any recent data. That process includes these steps if you use a tool like Azure Migrate and Modernize to replicate the server state. Otherwise, you might have to take manual steps, depending on how you staged the application.
- **Hydrate additional resources.** If you're using server state replication like with Azure Migrate and Modernize, you need to deploy Azure Virtual Machine instances as part of hydrating after the replication. There might be other items like load balancing rules that you weren't able to stage previously.
- **Turn off source servers.** If you still have source servers available after hydrating your resources, turn them off so that they don't interfere with your migration. If you need to revert back, you can turn on these servers after cleaning up the migration items.
- **Do isolated testing.** Do any testing that you can without transitioning users over to the migrated workload. This testing uses a test plan that's similar to the plan described in [Test your migration deployment in Azure](#).

- **Transition to the migrated workload.** Update your Domain Name System, connection strings, load balancing, and other items so that when users or systems try to access the application, they access its new location.
- **Do promotion testing.** Run your business testing plan again to confirm that everything works as expected.
- **Seek final approval.** Seek a final go/no-go decision for the workload with stakeholders.
- **Resume monitoring.** Enable any disabled monitoring and confirm that the environment is in an acceptable state.
- **Communicate that the promotion was successful.** Tell all necessary parties that the promotion is finished and that there are no forthcoming changes.

Next step

[Optimize cost after migration](#)

Feedback

Was this page helpful?



Optimize cost after migration

Article • 04/10/2024

After you migrate your workloads to Azure, you should optimize costs to ensure that you don't overspend. This article provides guidance on how to optimize your costs after migration and how to decommission retired assets with minimal business interruptions.

Optimize migrated workloads for cost

After you migrate your workloads and decommission unneeded resources, you can save on costs by optimizing your workload based on its live data.

You can resize workloads based on their performance during an assessment, but you might find while the workload is running in Azure that there are additional cost savings available.

Tools for optimizing costs

After you migrate to Azure, you have new tools available to manage your resource costs. Use the following list to help manage your cloud spend.

[+] Expand table

Tool	Description	Resource
Rightsize assets	Review the service usage metrics and rightsize them to match the workload requirements.	<ul style="list-style-type: none">Azure Advisor cost recommendations
Azure Reserved Virtual Machine Instances	Reserved instances let you commit to resources in Azure that run frequently. Consider reserving instances for workloads that are always active.	<ul style="list-style-type: none">Manage reservations for Azure resourcesAzure virtual machine (VM) sizing for maximum reservation usage
Azure savings plans	Azure savings plans provide savings up to 65% compared to pay-as-you-go pricing when you commit to spending a fixed hourly amount on compute services for one or three years.	<ul style="list-style-type: none">Azure savings plans recommendations
Cost management	You can use Microsoft Cost Management to monitor and manage the costs of the environment.	<ul style="list-style-type: none">Cost ManagementReservation recommendations in Advisor

Tool	Description	Resource
Financial operations documentation	Financial operations is a discipline that combines financial management principles with cloud engineering and operations to provide organizations with a better understanding of their cloud spending.	<ul style="list-style-type: none"> • What is financial operations?

Decommission retired assets

After you promote a migrated workload to production, the assets that ran the workload are no longer required and are considered out of service. But these assets still consume electricity and other resources which increases costs. Therefore, it's a good idea to shut down and dispose of retired assets to reduce expenses.

Shutting down and disposing of old assets and equipment might seem straightforward but unexpected problems can occur. Here are some tips on how to safely shut down and dispose of old resources without causing any problems for your business.

Continue monitoring

After you promote a migrated workload to production, you should continue to monitor the assets that are scheduled for retirement to ensure that production traffic is correctly routed.

While assets might be turned off, they might still utilize storage, network, and other infrastructure resources. If they're turned back on, they could cause unexpected problems unless they've been removed.

Monitor the following signals for the resources:

- **Compute:** Resource compute usage, like CPU and RAM.
- **Storage:** Resource storage usage, like disk input/output (I/O).
- **Network:** Resource network usage that includes inbound and outbound networking from appliances. For example, inspect assets that use firewalls and load balancers for communication.
- **Logs:** Windows and application logs.
- **Other signals:** Any other signals that you used to monitor the assets when they were hosted in their previous production environment.

In some migrations, assets aren't turned off. Instead, they're duplicated. Sudden spikes or even consistent moderate usage of infrastructure signals, along with network activity or new logs, can indicate that the asset is still in use.

Testing windows and dependency validation

Even with the best planning, production workloads might still contain dependencies on assets that are presumed retired. In such cases, turning off a retired asset could cause unexpected system failure. As such, treat the termination of any assets with the same care as system maintenance activity.

Establish proper testing and outage windows to facilitate the termination of the resource. You need a maintenance window to successfully test your assets before termination. Choose a period of time when you can test the assets without causing any business interruptions.

Define a testing and maintenance window

- **Low-impact times:** Identify a low-impact time for your testing window. Choose a time when application use is at its lowest.
- **Clear test cases:** Identify clear test cases that you can perform during the testing window that match real activities performed by users of the application. These activities shouldn't be surface level but should instead map out every process used. You can reuse the test cases from your migration if you have them. If you have users or other team members who frequently work in the application, try having them perform the tests.
- **Schedule and communicate:** Schedule a maintenance window for as long as you have available. You should aim for a minimum of four hours.
 - **Schedule:** Plan the window so that application users can plan ahead. Two weeks is reasonable.
 - **Communicate:** Announce the change in advance. Set the expectation that there might be an outage during this maintenance window and that the system might not be responsive. Users shouldn't expect the application to be available during this time.

Before the maintenance window

- **Perform test cases:** Run through the test cases and monitor any usage of the resources.
 - **If you discover usage,** you shouldn't proceed with the maintenance window. Instead, you should investigate further to see if the assets are still in use.
 - **If you don't discover usage,** you can proceed with the maintenance window.

During the maintenance window

- **Disable assets:** Disable the assets flagged for decommission.
 - Power the assets off if they're still powered on.
 - Remove the assets from any load balancers and confirm they aren't capable of responding to incoming requests.
- **Perform tests:** Perform your test cases against the workload that runs in Azure.
 - **Tests succeeded without failure:** The assets aren't in use at this time.
 - Communicate an end to the change window so that users know that they can expect stability in the application again.
 - Proceed to the next section after the tests succeed.
 - **Tests failed:** The assets might be in use at this time and more testing is necessary.
 - Re-enable the assets flagged for decommission and repeat the failed test cases.
 - If the test cases continue to fail, then there might be an unrelated problem. You need to test more within the maintenance window and should also begin escalation to ensure that you have the right level of support.
 - If the test cases stop failing, then the problem is likely related. You should leave the assets enabled and close out the maintenance window after completing testing.
 - Investigate the problem outside of the scheduled maintenance window. Schedule another maintenance window for changes to the migrated workload, and schedule extra maintenance windows for testing.

Holding period and data validation

After you complete your testing window, all assets flagged for decommission should be powered off and disconnected so you can operate the workload. You can proceed to the next phase of decommissioning, but don't immediately dispose of the assets.

Consider a holding period

It's not uncommon for migrations to miss data during replication processes. This is especially true for older data that isn't accessed regularly. Keep a retired asset for a while to serve as a temporary backup of the data. You should allow at least 30 days for holding and testing before disposing of retired assets.

Consider data governance requirements

Your organization's data governance team might have more requirements beyond a 30-day holding period.

- **Understand holding period obligations:** You should check with the necessary teams to understand the obligation for holding on to information and build a validation checklist for your specific legal requirements.
 - Having the asset operational isn't important at this time. Instead, the data on the information should be retrievable. Keep disks or backups to restore the data if necessary.
 - For example, if you have a SQL database server in your physical datacenter, you can back up its data and maintain it as a recoverable resource. Then you can decommission the virtual machine and set a holding time to retire the backup.

Next step

The migration is complete after you decommission the retired assets. This creates a good opportunity to improve the migration process with a retrospective to learn and improve.

[Conduct retrospectives](#)

Feedback

Was this page helpful?

 Yes

 No

Build a growth mindset with retrospectives

Article • 04/10/2024

Retrospectives reinforce the principles of a growth mindset: experimentation, testing, learning, sharing, growing, and empowering. They also provide a safe place for team members to share the challenges they faced in the current project and empower the team to create sustainable growth.

Retrospective structure

During retrospective meetings for the migration project, each member of the team is expected to share their thoughts regarding three basic questions:

- What went well?
- What could have been better?
- What did we learn?

Lessons learned

The retrospective marks the end of a release or iteration. As the team gains experience and learns lessons, they adjust the release and iteration backlog to reflect new processes and experiments to test. This begins the next iteration through the migration processes.

Teams that manage the next migration wave should apply the lessons they've learned to continually improve the migration.

Conduct a retrospective

You need a few things to get started with your migration retrospective:

- **An organizer:** A team member to act as an organizer or coach throughout the process. This team member focuses on guiding the team through the retrospective process and ensures that all ideas are heard and captured.
- **A method to track and organize:** A method to track and organize ideas in a visual way. This can be a physical whiteboard with sticky notes, a virtual whiteboard, or a DevOps tool.
- **Scoped questions:** You want to organize your tracking system with three questions:

- What went well?
- What could have been better?
- What did we learn?

After you complete the preceding steps, do the following steps with your team:

- **Individual answers:** Have each team member answer the question "*What went well?*" and document their answers.
 - If using a physical board, let each team member fill in their answers on the board.
 - If using a digital board, let each member fill in their answers via the tool.
- **Organizer read-out:** After each team member documents their thoughts, the organizer should review the feedback and then read it aloud to the team.
 - You should group together similar ideas as you discover them so that you recognize reoccurring trends.
 - Create labels or *parent* items as needed to help organize these thoughts.
 - If you find misplaced items that belong in another area, move them over at this time.
- Repeat this process for "*What could have been better?*".
 - During this process, encourage people to discuss their specific challenges with people, process, and technology. This should be an open space.
 - Try to determine what created delays with migrations, affected the release, or lead to workloads being pushed out of this sprint.
 - Think about how you can prevent these problems as you discuss the question "*What did we learn?*".

After you capture and group ideas from these questions, ask the team, "*What did we learn?*". Then brainstorm ideas for improvement that you can apply to the next sprint.

Example retrospective output

Consider the following example retrospective output:

[\[+\] Expand table](#)

What went well?	What could have been better?	What did we learn?
Communication with the business users for Application X and Application Y. (5 votes)	Involvement of developers to validate the network architecture for Application X and Application Z. (3 votes)	We need to review all firewall traffic from workloads before migration, even if we believe that

What went well?	What could have been better?	What did we learn?
		they have no specialty rules. (4 votes)
Development teams for Application Y were highly involved. (2 votes)	Understanding the networking requirements for Application X. (2 votes)	We need to make sure development teams know what's expected from them as part of the migration process. (4 votes)
Enablement of Azure Backup as part of migration process.	Understanding the networking requirements for Application Y. (1 vote)	We need to plan extra time for large SQL databases. (3 votes)

Next step

You should continue to implement the Cloud Adoption Framework for Azure's Manage methodology to ensure that you can continuously maintain and optimize your workloads in Azure.

[Cloud Adoption Framework - Manage methodology](#)

Feedback

Was this page helpful?

 Yes

 No

Review product migration scenarios

Article • 04/10/2024

Specific migration scenarios require different approaches and tools. This article is a collection of links to migration guidance for specific products and services.

Amazon Web Services (AWS)

[+] Expand table

Link	Description
Discover AWS instances	Complete a tutorial to learn how to discover Amazon Web Services (AWS) instances using the Azure Migrate: Discovery and assessment tool .
Assess AWS instances for migration to Azure	Learn how to assess AWS instances for migration to Azure using the Azure Migrate: Discovery and assessment tool.
Migrate AWS instances	Complete a tutorial to learn how to discover and assess AWS virtual machines (VMs), and then migrate them to Azure VMs using the Azure Migrate: Discovery and assessment tool and the Migration and modernization tool.

Containers

[+] Expand table

Link	Description
Migrate to Azure Kubernetes Service	Get the current recommended Azure Kubernetes Service configuration to help you plan and carry out a successful migration to Azure Kubernetes Service.

Windows and Linux

VMware servers

[+] Expand table

Link	Description
Discover servers running in a VMware environment	Learn how to discover the servers that run in your VMware environment by using the Azure Migrate discovery and assessment tool.
Assess VMware VMs for migration to Azure VMs	Use the Azure Migrate discovery and assessment tool to assess discovered servers in your VMware environment in preparation for migration to Azure VMs.
Select a VMware migration option	Learn how you can migrate VMware VMs to Azure by using the migration and modernization tool.
Move on-premises VMware infrastructure to Azure VMware Solution	Learn how to use Azure VMware Solution, a first-party Microsoft offering that's backed by VMware, to create a private cloud in Azure. The cloud has native access to VMware vCenter Server and other VMware tools that support workload migrations.

Hyper-V servers

[\[+\] Expand table](#)

Link	Description
Discover servers running in a Hyper-V environment	This tutorial shows you how to use the Azure Migrate discovery and assessment tool to discover servers that run in your Hyper-V environment.
Assess Hyper-V VMs for migration to Azure	Learn how to use the Azure Migrate discovery and assessment tool to assess discovered servers in your Hyper-V environment for migration to Azure.
Migrate Hyper-V VMs to Azure	This article demonstrates how to assess Hyper-V machines for migration to Azure and how to migrate them.

Remote Desktop Services and Azure Virtual Desktop

[\[+\] Expand table](#)

Link	Description
Move an on-premises Remote Desktop Services instance to Azure Virtual Desktop	Learn how to migrate and modernize an on-premises virtual desktop infrastructure (VDI) environment by moving an instance of Remote Desktop Services in Windows Server to Azure Virtual Desktop.

Databases and data platforms

SQL Server

[+] Expand table

Link	Description
SQL Server migration guides	Get the guidance you need to migrate to the Azure SQL family of SQL Server database engine products in the cloud: Azure SQL Database, Azure SQL Managed Instance, and SQL Server on Azure Virtual Machines.
Migrate from SQL Server to Azure SQL	Learn how to migrate your SQL Server instance to Azure SQL Database.
Migrate from SQL Server to Azure SQL Managed Instance	Learn how to migrate your SQL Server instance to Azure SQL Managed Instance.
Migrate from SQL Server to SQL Server on Azure Virtual Machines	Learn how to discover and assess your user databases for migration from SQL Server to SQL Server on Azure Virtual Machines, and then migrate the databases by using tools and techniques based on your requirements.

IBM Db2

[+] Expand table

Link	Description
Migrate from IBM Db2 to Azure SQL Database	Learn how to migrate your IBM Db2 databases to Azure SQL Database by using SQL Server Migration Assistant (SSMA) for IBM Db2.
Migrate from IBM Db2 to Azure SQL Managed Instance	Learn how to migrate your Oracle schemas to Azure SQL Managed Instance by using SSMA for Oracle.
Migrate from IBM Db2 to Azure SQL on Azure Virtual Machines	Learn how to migrate your Oracle schemas to SQL Server on Azure Virtual Machines by using SSMA for Oracle.

Oracle

[+] Expand table

Link	Description
Migrate from Oracle to Azure SQL Database	This article teaches you how to migrate your Oracle schemas to Azure SQL Database by using SSMA for Oracle.
Migrate from Oracle to Azure SQL Managed Instance	This article teaches you how to migrate your Oracle schemas to Azure SQL Managed Instance by using SSMA for Oracle.
Migrate from Oracle to Azure SQL on Azure Virtual Machines	Learn how to migrate your Oracle schemas to SQL Server on Azure Virtual Machines by using SSMA for Oracle.

Microsoft Access

[\[+\] Expand table](#)

Link	Description
Migrate from Microsoft Access to Azure SQL Database	Learn how to migrate your Access database to Azure SQL Database by using SSMA for Access.

Linux and open-source databases

[\[+\] Expand table](#)

Link	Description
Migrate open-source databases to Azure	Learn how to migrate open-source workloads from PostgreSQL and MySQL databases to the equivalent services in Azure.
Migrate MySQL databases to Azure SQL	Learn how to migrate your MySQL database to Azure SQL Database by using SSMA for MySQL.
Migrate PostgreSQL databases to Azure	You can use Azure Database Migration Service to migrate databases from an on-premises PostgreSQL instance to Azure Database for PostgreSQL with minimal downtime to the application.
Migrate Azure Database for MariaDB to Azure Database for MySQL ↗	Azure Database for MariaDB is being retired. Learn how to migrate to Azure Database for MySQL.

Storage migration

[\[+\] Expand table](#)

Link	Description
Storage migration overview	This article focuses on storage migrations to Azure, including the migration of unstructured data and block-based devices such as disks and storage area networks (SANs).
Tools for unstructured storage migration	This article offers a comparison matrix of basic functionality in different tools that you can use for the migration of unstructured data.

Java applications

[\[+\] Expand table](#)

Link	Description
Migration overview	This migration guidance covers mainstream Java on Azure scenarios and provides high-level planning suggestions and considerations.
Migrate Spring Boot applications to Azure Spring Apps	This article describes what you should be aware of when you migrate an existing Spring Boot application to run on Azure Spring Apps.
Migrate Spring Cloud applications to Azure Spring Apps	This article describes what you should be aware of when you migrate an existing Spring Cloud application to run on Azure Spring Apps.
Migrate WildFly applications to WildFly on Azure Kubernetes Service	This article describes what you should be aware of when you migrate an existing WildFly application to run on WildFly in an Azure Kubernetes Service container.

Apache Tomcat to Azure

[\[+\] Expand table](#)

Link	Description
Migrate Apache Tomcat applications to Tomcat on Azure App Service	This article describes what you should be aware of when you migrate an existing Tomcat application to run on Azure App Service by using Tomcat 9.0.
Migrate Tomcat applications to Azure Container Apps	This article describes what you should be aware of when you migrate an existing Tomcat application to run on Azure Container Apps.
Migrate Tomcat applications to containers on Azure	This article describes what you should be aware of when you migrate an existing Tomcat application to run on Azure

Link	Description
Kubernetes Service	Kubernetes Service.
Java web app containerization and migration to Azure Kubernetes Service	In this article, learn how to containerize Java web applications (running on Apache Tomcat) and migrate them to Azure Kubernetes Service by using the Azure Migrate: App Containerization tool.

Oracle WebLogic Server to Azure

[Expand table](#)

Link	Description
Migrate WebLogic Server applications to Azure Kubernetes Service	This article describes what you should be aware of when you migrate an existing Oracle WebLogic Server application to run on Azure Kubernetes Service.
Migrate WebLogic Server applications to Azure Virtual Machines	This article describes what you should be aware of when you migrate an existing WebLogic Server application to run on Azure Virtual Machines.
Migrate WebLogic Server applications to WildFly on Azure Kubernetes Service	This article describes what you should be aware of when you migrate an existing WebLogic Server application to run on WildFly in an Azure Kubernetes Service container.
Tutorial: Manually install WebLogic Server on Azure Virtual Machines	This tutorial shows the steps to install WebLogic Server and configure a WebLogic Server cluster on Azure Virtual Machines on Windows or GNU/Linux.
Tutorial: Migrate WebLogic Server to Azure Kubernetes Service within a custom virtual network	This tutorial shows you how to deploy the offer to integrate WebLogic Server with Azure Kubernetes Service by using a custom virtual network in the consumer's subscription.
Tutorial: Migrate a WebLogic Server cluster to Azure by using Azure Application Gateway as a load balancer	This tutorial walks you through the process of deploying WebLogic Server by using Azure Application Gateway.
Tutorial: Migrate WebLogic Server to Azure Virtual Machines with high availability and disaster recovery	This tutorial shows you a simple and effective way to implement high availability (HA) and disaster recovery (DR) for Java by using WebLogic Server on Azure Virtual Machines
Migrate WebLogic Server applications to JBoss EAP on Azure App Service	This article describes what you should be aware of when you migrate an existing WebLogic Server application to run on Azure App Service by using Red Hat JBoss Enterprise Application Platform (JBoss EAP).

IBM WebSphere to Azure

[+] Expand table

Link	Description
Migrate WebSphere applications to Azure Kubernetes Service	This article describes what you should be aware of when you migrate an existing IBM WebSphere Application Server workload to IBM WebSphere Liberty or Open Liberty on Azure Kubernetes Service.
Migrate WebSphere applications to Azure Red Hat OpenShift	This article describes what you should be aware of when you migrate an existing WebSphere Application Server workload to IBM WebSphere Liberty or Open Liberty that runs on Azure Red Hat OpenShift.
Migrate WebSphere applications to Azure Virtual Machines	This article describes what you should be aware of when you migrate an existing WebSphere Application Server traditional application to run on Azure Virtual Machines.
Tutorial: Manually install IBM WebSphere Application Server Network Deployment traditional on Azure Virtual Machines	This tutorial shows you how to install traditional IBM WebSphere Application Server Network Deployment traditional and configure a WebSphere Application Server cluster on Azure Virtual Machines on GNU/Linux.
Migrate WebSphere applications to JBoss EAP on Azure App Service	Learn what you should be aware of when you migrate an existing WebSphere application to run on Azure App Service by using JBoss EAP.
Migrate WebSphere applications to WildFly on Azure Kubernetes Service	This article describes what you should be aware of when you migrate an existing WebSphere application to run on WildFly in an Azure Kubernetes Service container.

JBoss EAP to Azure

[+] Expand table

Link	Description
Migrate JBoss EAP applications to JBoss EAP on Azure App Service	This article describes what you should be aware of when you migrate an existing JBoss EAP application to run on JBoss EAP in an Azure App Service instance.
Migrate JBoss EAP applications to JBoss EAP on Azure Virtual Machines	This article describes what you should be aware of when you migrate an existing JBoss EAP application to run on JBoss EAP on Azure Virtual Machines.

Link	Description
Migrate JBoss EAP applications to Azure Red Hat OpenShift	This article describes what you should be aware of when you migrate an existing JBoss EAP application to run on Azure Red Hat OpenShift.
Migrate JBoss EAP applications to WildFly on Azure Kubernetes Service	This article describes what you should be aware of when you migrate an existing JBoss EAP application to run on WildFly in an Azure Kubernetes Service container.

SAP

[\[+\] Expand table](#)

Link	Description
Migrate an SAP platform to Azure	SAP workloads require different tools and processes to replicate and deploy their assets than what Azure Migrate traditionally offers. Learn more in this adoption scenario.
Learning path for migrating SAP workloads to Azure	These Microsoft Learn training modules explore migrating SAP workloads to Azure, including databases that are larger than 20 terabytes (TB).
SAP on Azure migration ↗	Learn how to migrate your SAP estate to Azure, with resources to support your journey at every step.
Migrate from SAP ASE to Azure SQL	Learn how to migrate your SAP Adaptive Server Enterprise (ASE) databases to Azure SQL Database by using SSMA for SAP ASE.

Azure Local

[\[+\] Expand table](#)

Link	Description
Overview of an Azure Migrate-based migration for Azure Local	Get an overview of how to migrate Hyper-V VMs to your Azure Local instance by using Azure Migrate.

Azure Synapse Analytics

[\[+\] Expand table](#)

Link	Description
Design and performance considerations for Teradata migrations	Learn how to migrate from Teradata to Azure Synapse Analytics.
Design and performance considerations for IBM Netezza migrations	Learn how to migrate from IBM Netezza to Azure Synapse Analytics.
Design and performance considerations for Oracle migrations	Learn how to migrate from Oracle to Azure Synapse Analytics.

Multitenant migration

[\[\] Expand table](#)

Link	Description
Manage Azure Migrate projects at scale by using Azure Lighthouse	Get an overview of how Azure Lighthouse can help you use Azure Migrate in a scalable way across multiple Microsoft Entra tenants.

Feedback

Was this page helpful?

 Yes

 No

Modernize in the cloud

Article • 10/17/2022

Modernization is when you enhance workloads and the processes to support those workloads. It's all about maximizing value and adopting cloud technologies that allow you to unlock more benefits of the cloud. The goal of modernization is to make the cloud work for you. It can improve operational efficiency, reduce management overhead, and optimize costs. Modernization allows your business or organization to be more productive with less. Efficiency at this scale should be a priority.

We recommend modernizing in two phases:

- **Phase 1. Business alignment**
- **Phase 2. Modernization strategies**

Below you'll find an overview of each phase and guidance on where to start your modernization journey. We developed a framework that walks you through the most effective modernization processes.

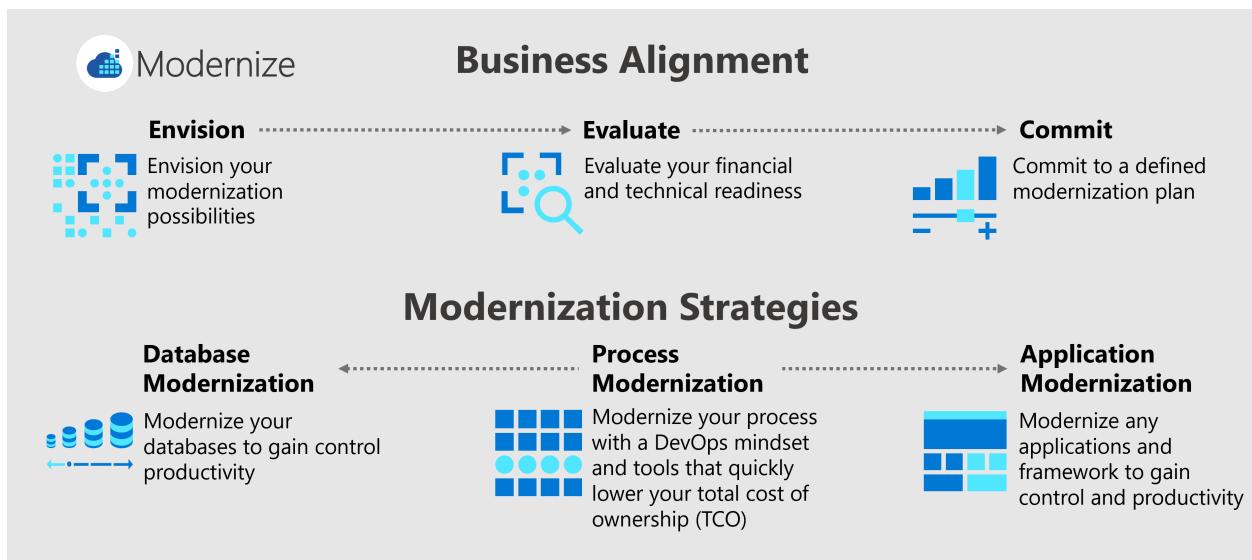


Figure 1: Overview of the modernization process

Phase 1. Business alignment

Phase 1 of modernization is where you identify your business goals and create a modernization roadmap to reach those goals. The roadmap will list the workloads you need to modernize and the modernization strategies you need to use.

When you finish creating your modernization roadmap, you can move on to phase two and begin modernizing.

Phase 2. Modernization strategies

Phase 2 is where you implement modernization strategies. Technical change happens in this phase. You'll adopt new methodologies and new technologies to enhance your processes, applications, and databases.

- **Process modernization** - We recommend adopting a DevOps methodology. DevOps will help speed up your modernization efforts and drive down your total cost of ownership. Process modernization is so key to workload modernization, it should be a necessary prerequisite to workload modernization.
- **Application and Database modernization** - On the technical side, we recommend adopting platform-as-a-service solutions (PaaS) in your modernization effort. Application and database PaaS technologies are cost efficient solutions that help you scale and reduce your management overhead.

Next steps

Modernization starts with business alignment.

[Learn about business alignment](#)

Business alignment for cloud modernization

Article • 09/12/2022

Modernization is a business decision. The goal is to improve your technology so that your business can scale. Since business goals are the key drivers of modernization, it's essential that your modernization efforts align with your business goals.



Diagram 1: Overview of the business alignment process

Definition of business alignment

Business alignment is the process of identifying your modernization motivations and creating a modernization roadmap around these motivations. It's that simple. You identify specific motivations for each workload you want to modernize. The motivations determine what modernization strategy you take.

Business alignment process

The business alignment process is three steps. It helps ensure that your modernization journey leads to the most profitable outcome. These steps are logical and sequential.

1. *Envision your modernization possibilities* – You align your modernization goals with the workloads that will help you reach those goals.
2. *Evaluate your cloud modernization readiness* – You assess the financial and technical readiness of each workload you want to modernize.
3. *Commit to the modernization path you choose* – You finalize your modernization roadmap for each workload and commit the resources required to make the mission a success.

These steps are actionable and give you a framework to align your modernization efforts and your business goals.

Next step

Envision your cloud modernization possibilities

Envision cloud modernization possibilities

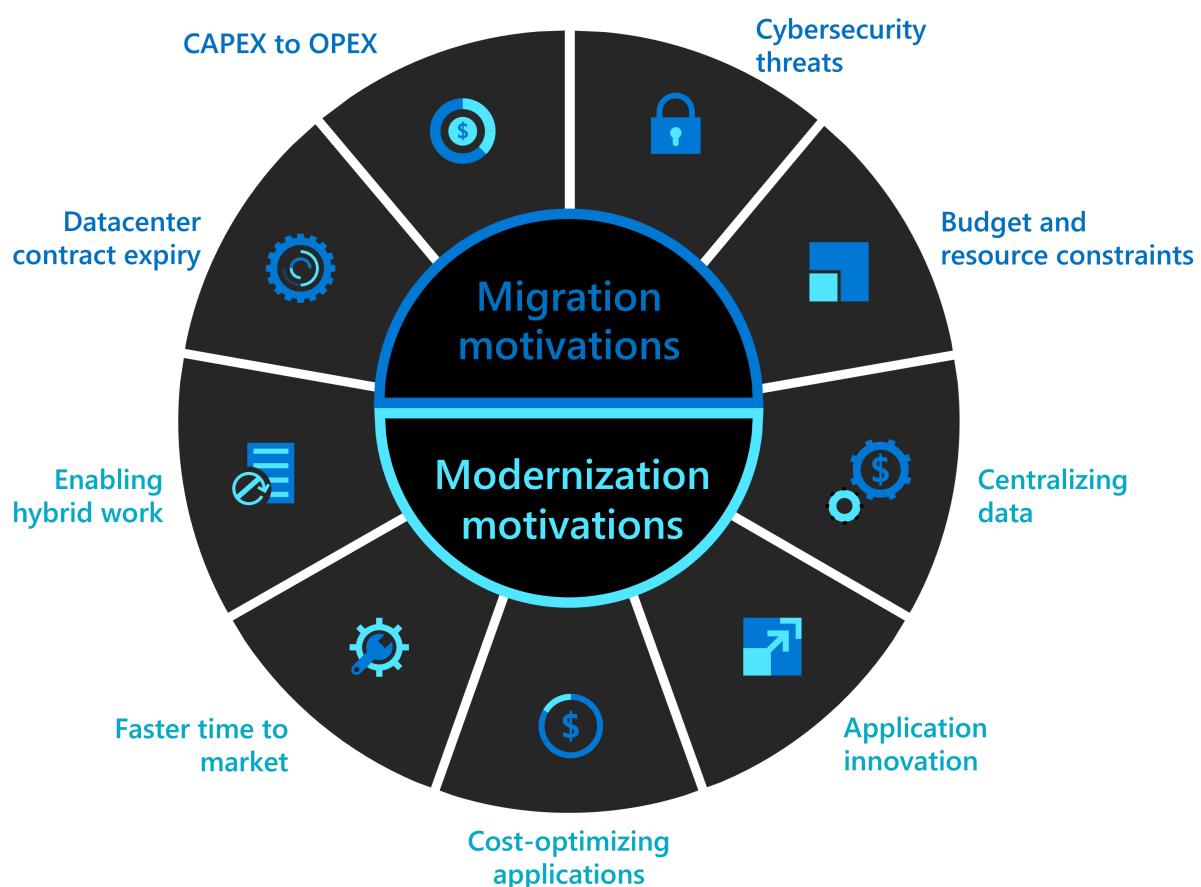
Article • 09/08/2022

Envisioning your cloud modernization possibilities starts with understanding your motivations. You have reasons for wanting to adopt the cloud, and these reasons are the key to modernization. Knowing your motivations lets you determine what you should modernize.

Follow these three steps to envision your cloud modernization possibilities.

Step 1 - Identify modernization motivations

Knowing *why* you're modernizing helps you determine *what* to modernize. So, make sure you have a good understanding of your motivations before moving forward. Your motivations for modernization will be unique to your business goals, but they likely aren't unique to you.



We see similar motivations across businesses and industries. Some of the most common modernization motivations are:

- *Enabling hybrid work:* Make applications accessible and secure for remote productivity.
- *Getting to market faster:* Increase deployment speed to see a faster return-on-investment.
- *Optimizing cost of applications:* Gain operational efficiencies to drive down the cost of ownership.
- *Innovating applications:* Adopt new technologies and architectures to enable rapid innovation.
- *Centralizing data:* Store application data in a centralized repository to enhance security, reduce redundancy, and drive transparency.

Step 2 - Identify workloads

Identify the workloads you need to modernize to fulfill your modernization motivations. A [workload](#) is a collection of IT assets (infrastructure, applications, and data) that support a business function. Data centers encourage you to manage IT assets separately.

The cloud lets you focus on workloads instead of separate IT assets. Modernizing infrastructure, applications, and data together as workloads lets you streamline your modernization processes. You can reuse these processes and apply them to each workload requiring the same modernization strategy.

How to identify workloads:

Identify workloads by making a list of business functions. Business functions are the key components of your business.

Identify the IT assets that support each business function. The group of IT assets that support a specific business function is a workload (*see table for simplified examples*).

Business Function	Workload
E-commerce	Web App
Website	Business logic
	Database
	Servers
	Payment system

Business Function	Workload
Factory Production	Machine/Robot IOT device Programmable logic controller Monitoring Server

Step 3 - Align modernization motivations and workloads

Align your modernization motivations and workloads so you can evaluate their readiness in the next step.

1. *Map motivations and workloads.* Map your modernization motivations from step 1 above to the workloads you identified in step 2.
2. *Assign one motivation to each workload.* Assigning a single modernization motivation to each workload will help you organize your work when it's time to modernize (see table below for examples).
3. *When a single motivation is unclear, conduct a well-architected review on the workload.* The well-architected review will help you figure out what your motivation should be. After running the review, return to this step and complete step 3.

Business function	Workload	Motivation
E-commerce website	Web app Business logic Database Servers Payment System	Application innovation
Factory production	Machine/Robot IOT device Programmable Logic Controller Monitoring Server	Centralizing data

Next steps

When you have a single motivation for each workload, you can evaluate your options.

Evaluate your cloud modernization readiness

Evaluate cloud modernization readiness

Article • 10/24/2022

Once you've envisioned your cloud modernization possibilities, you can evaluate whether you're ready to modernize in the cloud.

Evaluating your readiness for modernization has two components. You should assess your financial readiness and then your technical readiness.

- *Financial readiness* - helps you determine if you need more time to review your applications.
- *Technical readiness* - helps you determine if you're ready to modernize or need to explore a different path.

Step 1 - Evaluate the financial readiness of each workload

A financial readiness assessment helps you determine if it makes business sense to modernize your workload. Financial motivations are the key drivers of cloud modernization. Cloud modernizations can improve margins through efficiencies and generate new revenue streams. To realize those benefits, you need to evaluate the financial aspects of the effort.

For each workload you want to modernize, answer the following questions to assess your financial readiness.

- *Can you quantify the business value of modernizing the workload?*
- *Do you know what your modernization cost will be?*
- *Are these workloads business-critical?*
- *Does the cost of modernization meet your desired cost savings?*

Answering yes to all questions means you're likely ready to modernize. If you answered *no* to any of the questions, we recommend you conduct an [Azure Well-Architected Review](#) of your workload.

Step 2 will help you determine if you're ready to modernize or need to explore another adoption path.

Step 2 - Evaluate the technical readiness of each workload

A technical readiness assessment helps you determine if your workload is ready for modernization or is better fit for a different cloud adoption strategy.

For each workload you want to modernize, answer the following questions to begin assess your technical readiness.

Question	Yes	No
<i>Do you have enough control over the workload to modernize it?</i>	Modernize	Migrate
<i>Is your business actively investing in these workloads?</i>	Modernize	Replace
<i>Will these modernized workloads need to operate in hybrid or multicloud environment?</i>	Modernize	Migrate
<i>Are your workloads portable?</i>	Modernize	Migrate
<i>Do you plan to keep the current architecture?</i>	Modernize	Conduct Azure Well-Architected Review

If you answered yes to all the technical-readiness questions, you're likely ready to modernize the workload.

Next steps

You've [envisioned](#) and [evaluated](#). It's time to commit to a modernization path.

[Commit to modernization](#)

Commit to modernization

Article • 11/17/2022

Modernization requires commitment. It takes time and money to modernize. Creating a modernization roadmap can accelerate approval from the right stakeholders. These stakeholders are typically business leads and technical leads. You'll need to identify the stakeholders, define their involvement, and get the commitment you need.

- *Technical Leads* - Technical leads need to own the modernization roadmap and present it to the business leads for investment.
- *Business leads* - Business leads should review the modernization roadmap, understand the ROI, and approve the modernization.

For more information, see:

- [Business conversations](#)
- [Business outcomes template](#)

Here are steps you can follow to prepare for and gain modernization commitment.

Identify modernization strategies

You need to figure out what modernization strategies help you meet your modernization goals. We covered five modernization motivations in the [envision step](#). These motivations will determine what modernization strategies you'll commit to.

We've provided a table that aligns motivations with two or more modernization strategies.

Modernization motivation (What's your goal?)	Modernization strategy (How to reach your goal)
<i>Enabling hybrid work</i>	<ul style="list-style-type: none">• Process modernization• Application modernization
<i>Faster time-to-market</i>	<ul style="list-style-type: none">• Process modernization• Application modernization
<i>Cost optimization</i>	<ul style="list-style-type: none">• Process modernization• Application modernization• Database modernization

Modernization motivation (What's your goal?)	Modernization strategy (How to reach your goal)
<i>Application innovation</i>	<ul style="list-style-type: none"> • Process modernization • Application modernization
<i>Centralizing data</i>	<ul style="list-style-type: none"> • Process modernization • Database modernization

Notice that *process modernization* is listed with every modernization motivation.

Modernizing your process is essential to working in the cloud. It's the fastest way to reduce the total cost of ownership (TCO). No matter your motivation for modernization, it's critical to modernize your process.

Add modernization strategies

With the modernization strategies identified, it's time to update the modernization update. your workload table with modernization strategies by adding the modernization strategies you identified in step 1 with the table you created in the [envision step of business alignment](#) (*see the following example*).

Business function	Workload	Motivation	Modernization Strategy
E-commerce website	Web app Business logic Database Servers Payment System	Application innovation	<ul style="list-style-type: none"> • Process modernization • Application modernization
Factory production	Machine/Robot IOT device Programmable Logic Controller Monitoring Server	Centralizing data	<ul style="list-style-type: none"> • Process modernization • Database modernization

Create a modernization timeline for each workload

Create a modernization timeline for each workload. The amount of time it takes to modernize your workloads depends on your team's experience and the complexity of

your situation (see table).

Business Function	Workload	Motivation	Modernization Strategy	Timeline
E-commerce Website	Web App Business logic Database Servers Payment System	Application innovation	<ul style="list-style-type: none">• Process modernization• Application modernization	N-weeks
Factory Production	Machine/Robot IOT device Programmable Logic Controller Monitoring Server	Centralizing data	<ul style="list-style-type: none">• Process modernization• Database modernization	N-weeks

You might have the technical expertise to estimate the modernization effort. If not, see the following resources to get the expert advice you need.

- [Azure Migration and Modernization Program ↗](#)
- [Fast Track for Azure ↗](#)

Commit resources to the modernization

Business leads provide the budget for the modernization effort. The budget should allow the technical leads to put the right people on the project and give the technical team the right tools.

Next steps

By completing the previous steps, you've created a high-level modernization roadmap. This roadmap identifies and assigns resources to the following business aspects:

- What you want to modernize (workload)
- Why you want to modernize (modernization motivation)
- How you can modernize (modernization strategies)
- When you'll modernize (timeline)

With your modernization roadmap and commitments in place, you can begin modernizing.

[Modernization strategies](#)

Cloud modernization strategies

Article • 09/08/2022

Modernization is a business decision that seeks to maximize value for your business. You achieve this value by enhancing your technology and refining the process to support that technology.

By now, you should have a clear understanding of your modernization goals and the strategies you need to take to achieve those goals. We created a modernization roadmap in the [business alignment phase](#) of the modernization framework. The roadmap will direct you to the modernization strategies that align with your modernization goals.

Definition of modernization strategy

A modernization strategy is a modernization effort that targets a key component of your business technology. We defined modernization as enhancing your workloads and the process to support those workloads. A modernization strategy gives you concrete steps to modernize both your workload and supporting process.



Modernization strategies

We scope modernization to just three different modernization strategies. These strategies provide the maximum value for your business with the least amount of effort.

- *Process modernization* - Adopt a DevOps methodology to modernize your development and operations. Process modernization is essential to lowering the total cost of ownership of your workloads.
- *Application modernization* - Adopt PaaS solutions to modernize any application or framework.

- *Database modernization* - Adopt database PaaS and infrastructure-as-a-service (IaaS) solutions to modernize any databases.

Process modernization is essential for getting the maximum value out of your modernization efforts. Start with process modernization first if you're not already using a DevOps methodology.

Next steps

[Modernize your process](#)

Modernize your processes for the cloud

Article • 05/19/2023

Process modernization creates the mechanisms in your business to gain operational efficiencies that lower the total cost of ownership of your workloads. We recommend adopting a DevOps methodology to modernize your processes.

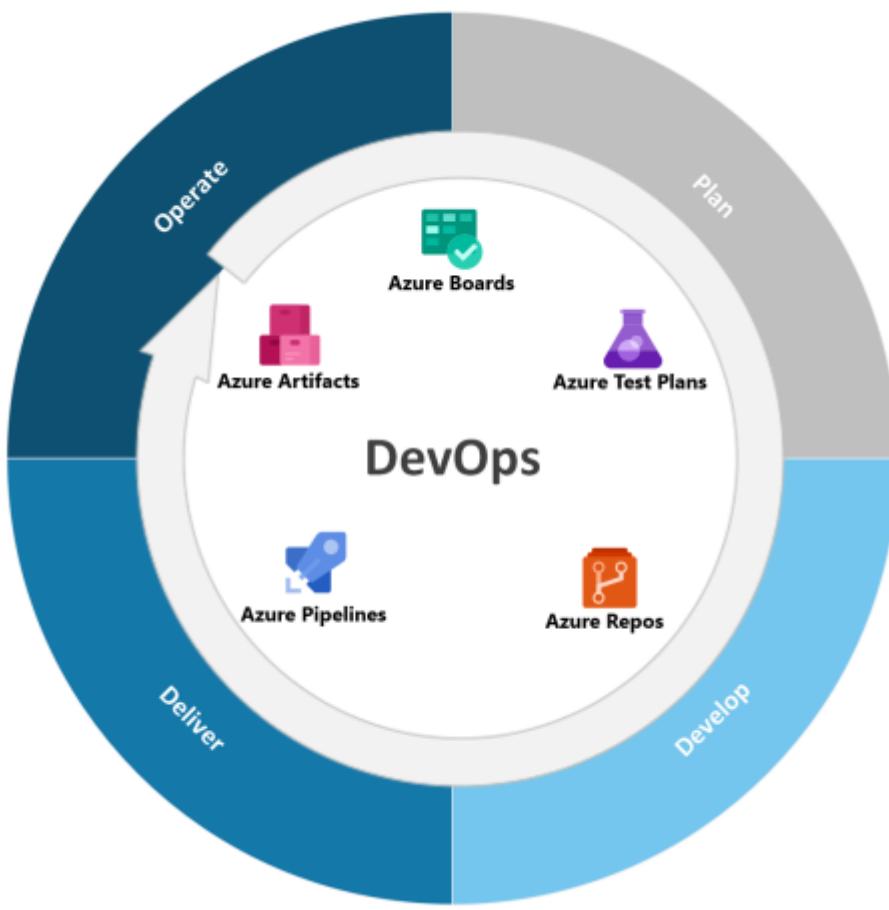


Diagram 1: High-level DevOps loop with DevOps tools

The DevOps methodology integrates planning, development, delivery, and operations into a single pipeline. Integration allows teams to name the highest priorities in your organization and resolve them together. The collective focus on the things that matter creates the most value with the least amount of work.

We've broken process modernization into three steps: tools, strategies, and benefits.

Step 1 - Adopt DevOps tools

Adopting DevOps tools can help you plan better and deliver value to customers faster.

In the following table, we've listed the DevOps tool and description. Azure has a set of DevOps tools called [Azure DevOps](#). You can use these Azure tools to implement DevOps, or you can use open-source solutions. For each DevOps tool in the table, we listed a service from the Azure DevOps Services and an alternative open-source solution.

DevOps tool	Description	Azure DevOps solution	Open-source solution
Source control	Code repositories for your project.	Azure Repos	GitHub
CI/CD pipeline	Continuously build, test, and deploy to any platform and cloud.	Azure Pipelines	Jenkins
Task board	Plan, track, and discuss work across your teams.	Azure Boards	TaskBoard
Package manager	Share code and publish packages	Azure Artifacts	Nexus
Test management	Run tests and quality assurance on code	Azure Test Plans	Selenium

With your DevOps tools in place, you're ready to modernize your planning strategies.

Step 2 - Adopt DevOps strategies

Modernize your process by adopting DevOps strategies. DevOps strategies are ways of doing things in an Agile framework. You integrate your development and operations, focus on the highest-priority tasks, and work on tasks for a defined period of time. These strategies allow you to modernize effectively in the cloud.

You're going to consult your modernization roadmap and plan your modernization efforts around it.

Consult your modernization roadmap

Consult the modernization roadmap you created in the [commit phase](#) to begin modernizing your planning strategies. It has all the details you need.

Here's the sample modernization roadmap we created.

Business function	Workload	Motivation	Modernization strategy	Timeline

Business function	Workload	Motivation	Modernization strategy	Timeline
E-commerce Website	Web app Business logic Database Servers Payment system	Application innovation	<ul style="list-style-type: none"> • Process modernization • Application modernization 	X-weeks

Plan your modernization efforts

Plan your modernization efforts using your modernization roadmap. Complete the steps in the following table to divide the work into manageable pieces and assign it to members of your team.

Step	Work tasks	Example
1. Identify major efforts	Major efforts include your modernization motivations and workloads. Major efforts are <i>epics</i> in Agile.	Application innovation for the E-commerce website.
2. Break down the major efforts (<i>epics</i>) into smaller tasks (<i>user stories</i>)	Analyze the major efforts (<i>epic</i>) from the perspective of your customers. Divide the major efforts (<i>epics</i>) into smaller, logical tasks that correspond to customer needs. Tasks created from epics are called <i>user stories</i> in Agile. The number of user stories you have for each epic depends on the complexity of the workload. There's no right or wrong number.	"As a customer, I want a more responsive website so I have a better shopping experience" "As a customer, I want a complete order history so I can reorder items easier"

Step	Work tasks	Example
3. Break down the tasks (<i>user stories</i>) further into smaller tasks (tasks)	These smaller tasks should be specific and detailed. Create as many as you need to meet the request in the user story. Subtasks are <i>tasks</i> in Agile.	Choose an App Service plan Configure resiliency Configure caching
4. Prioritize your user stories	Prioritize your user stories, so your team can meet the most important goals first.	1. Better shopping experience 2. Order history
5. Assign work (sprints)	Set a two-week period for your work efforts. These periods are <i>sprints</i> in Agile. Have your team members take on tasks that they can complete within the two-week period.	Assign two-week tasks (App Service plan, configure resiliency, configure caching) to your team.

This table captures the essentials of DevOps. It's enough to work with, but you might need more guidance. For more DevOps guidance, see [DevOps planning guidance](#).

Step 3 - Adopt DevOps benefits

You've adopted DevOps tools and adopted DevOps strategies. You're ready to adopt the benefits of DevOps. DevOps allows you to learn, improve, and scale faster. The time-bound iterations and team integration will enable your team to modernize better.

Next steps

You're ready to modernize your applications or databases. You should consult the modernization roadmap you created in [business alignment](#). The road map will let you know what to focus on next.

[Modernize your applications](#)

[Modernize your databases](#)

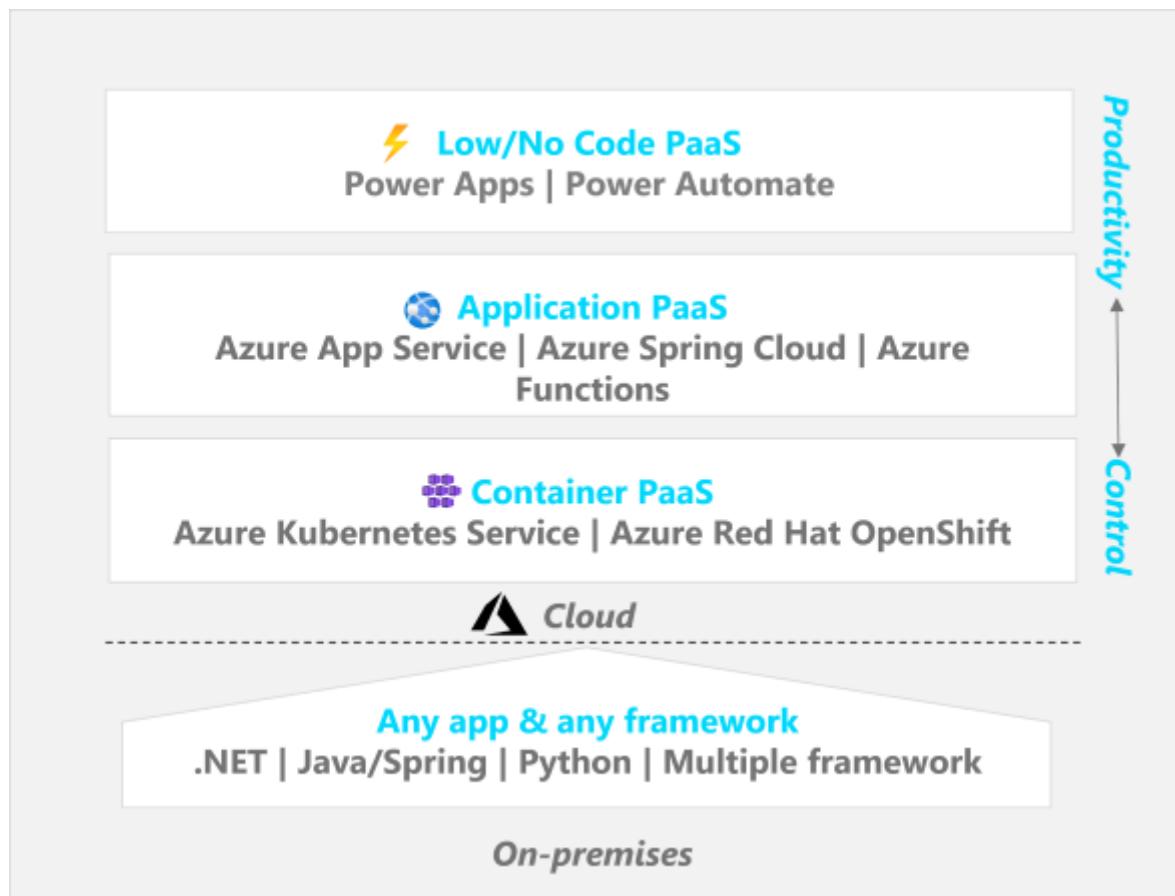
Modernize any application in the cloud

Article • 02/22/2024

Modernizing your applications can rapidly transform how people interact with your business or organization. The goal of application modernization is to enhance your applications to meet the needs of internal users and external customers. Adopting platform-as-a-service (PaaS) solutions lets you modernize any application or framework and enables your business to scale.

PaaS solutions:

PaaS gives you the flexibility to modernize any application or framework. Your applications running .NET, Java/Spring, Python, or multiple frameworks have a home in a PaaS solution. You can choose three variations of PaaS technologies depending on the balance of control and productivity you want.



- **Container PaaS** - Contains PaaS solutions that use container technologies to run your workloads. You provide code and manage agent nodes. The platforms manage health, maintenance, and deployment. Azure Kubernetes Service and Azure Red Hat OpenShift are examples of container PaaS solutions. They give you the most control over your workloads.

- *Application PaaS* - Application PaaS solutions use virtualization to run your workload without containers. You provide code and select configuration options. The platform manages health, availability, and deployment. There's less management than with Container PaaS. Azure App Service, Azure Spring Cloud, and Azure Functions are examples of application PaaS solutions. They give you a balance of control and productivity.
- *Low/No Code PaaS* - Low code PaaS and no-code PaaS solutions let you build apps with little to no coding required. You can rapidly build applications and focus more time on your business. Power Apps and Power Automate are low/no code PaaS solutions that maximize your productivity.

Step 1 - Use a decision tree to narrow options

Use a [decision tree](#) to narrow your modernization options. The goal is to narrow the options, not to decide which option is right for you. After you have your options, continue to step 2.

Step 2 - Find the right implementation guidance

Find the right implementation guidance. You'll have a general idea of the service you want to use after working through the decision tree. You still want to make sure the solution gives you the right balance of control and productivity. Finding the right balance is a main consideration in application modernization.

Use the following table to find a solution that meets your needs. When you find the right solution, follow the implementation guidance to modernize your application.

[] [Expand table](#)

Control vs. Productivity	Solution	Your needs	Implementation guidance
<i>Most control</i>	Azure Kubernetes Service (AKS) Azure Red Hat OpenShift	<ul style="list-style-type: none"> • Infrastructure control • Less administrative burden orchestrating clusters and nodes 	Azure Kubernetes Service (AKS) Azure Red Hat OpenShift
<i>Balance</i>	Azure App Service	<ul style="list-style-type: none"> • Focus on developing customer code 	Azure App Service

Control vs. Productivity	Solution	Your needs	Implementation guidance
	Azure Spring Apps	<ul style="list-style-type: none"> Automated infrastructure 	Azure Spring Apps
<i>Most productivity</i>	Power Apps Power Automate	<ul style="list-style-type: none"> Fastest time-to-market Applications and automation built with little to no coding experience 	Power Apps Power Automate

Next steps

Get more guidance with the [Azure Migration and Modernization Program](#) ↗

If you haven't already, modernize your databases.

[Modernize your databases](#)

Feedback

Was this page helpful?

 Yes

 No

Modernize any database in the cloud

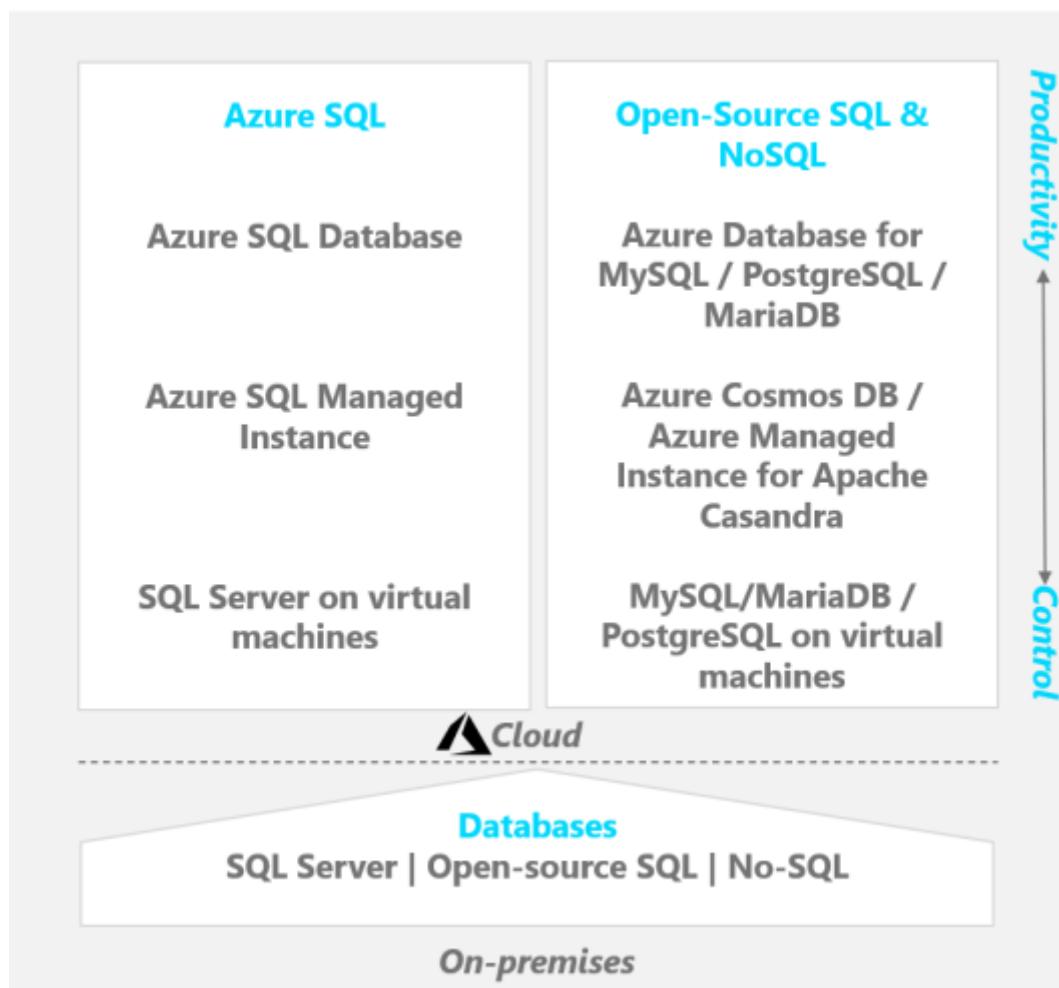
Article • 09/16/2022

Modernizing your databases can improve how you store, process, and feed data. The goal of database transformation is to improve the reliability, performance, and cost of your data. Adopt platform-as-a service (PaaS) and infrastructure-as-a-service (IaaS) solutions to modernize any database in the cloud.

Two categories of database solutions:

We have two categories of database solutions. These two categories allow you to modernize any type of database in the cloud.

- "*Azure SQL*" is a category of database solutions for SQL Server and other closed-source SQL databases.
- "*Open-source and NoSQL*" is a category of database solutions for open-source SQL databases and NoSQL databases.



PaaS and IaaS solutions:

There are two PaaS and one IaaS solution to choose from in each database category. Each option provides a different balance of control and productivity. We include infrastructure-as-a-service (IaaS) solutions to accommodate the different database origins customers start with.

Azure SQL:

- SQL Server on virtual machines (IaaS)
- Azure SQL Managed Instance (PaaS)
- Fully managed Azure SQL database (PaaS)

Open-source SQL & NoSQL:

- MySQL, MariaDB, or PostgreSQL on virtual machines (IaaS)
- Azure Cosmos DB and Azure Managed Instance for Apache Casandra (PaaS)
- Fully managed MySQL, MariaDB, and PostgreSQL databases (PaaS)

Use the tables below to find implementation guidance on modernizing your database.

Option 1 - Modernize your SQL Server and other SQL databases

Modernize your SQL Server and other SQL databases. Azure SQL supports SQL Server and other proprietary SQL databases like Oracle and Db2.

Finding the right balance of control and productivity is a main consideration in database modernization. Use the following table to find the right balance. Check to see if the solution listed meets your needs.

When you find the right solution, follow the implementation guidance to modernize your application.

Control vs. Productivity	Solution	Your needs	Implementation guidance
---------------------------------	-----------------	-------------------	--------------------------------

Control vs. Productivity	Solution	Your needs	Implementation guidance
<i>Most control</i>	SQL Server on Windows or Linux virtual machines (VMs)	<ul style="list-style-type: none"> • OS control • Quick modernization 	Database origin: SQL Server Oracle Db2 Azure Database Migration Service
<i>Balance control & productivity</i>	Azure SQL Managed Instance	<ul style="list-style-type: none"> • Near 100% compatibility with SQL Server (Enterprise Edition) • Automated patching • Native high availability • Instance-scoped features (Service Broker, SQL Server Agent, etc.) 	Database origin: SQL Server Oracle Db2
<i>Most productivity</i>	Azure SQL Database	<ul style="list-style-type: none"> • A multi-tenant SaaS application • Elasticity • To scale compute independent from storage 	Database origin: SQL Server Oracle Db2 Access SAP ASE

Option 2 - Modernize your open-source SQL & NoSQL databases

Modernize your open-source SQL databases and NoSQL databases. Finding the right balance of control and productivity is a main consideration in database modernization. Use the table to find the right balance. Check to see if the solution listed meets your needs.

When you find the right solution, follow the implementation guidance to modernize your application.

Control vs. Productivity	Solution	Your needs	Implementation guidance
<i>Most control</i>	Open-source databases on virtual machines - MySQL , MariaDB , PostgreSQL	<ul style="list-style-type: none"> • OS control • Zone redundancy 	Using the Azure Database Migration Service Installing a database on a VM
<i>Balance control & productivity</i>	Azure Managed Instance for Apache Cassandra	<ul style="list-style-type: none"> • Hybrid deployment • Automated deployment & scaling for Apache Cassandra datacenters • Automated patching • Automated health checks 	Apache Cassandra
<i>Most productivity</i>	Azure Cosmos DB	<ul style="list-style-type: none"> • A fully managed NoSQL database • Cost-efficiency • Support for MongoDB & Gremlin 	Azure Cosmos DB
<i>Most productivity</i>	Azure Database for MySQL	<ul style="list-style-type: none"> • A fully managed SQL solution • Cost and performance efficiency 	Azure Database for MySQL Using the Azure Database Migration Service
<i>Most productivity</i>	Azure Database for MariaDB	<ul style="list-style-type: none"> • A fully managed SQL solution • Cost and performance efficiency 	Azure Database for MariaDB
<i>Most productivity</i>	Azure Database for PostgreSQL	<ul style="list-style-type: none"> • A fully managed SQL solution • Cost and performance efficiency 	Azure Database for PostgreSQL

Next steps

Get more guidance with the [Azure Migration and Modernization Program](#) ↗

[Learn how to Innovate in the cloud](#)

Cloud adoption-related innovation

Article • 10/14/2024

All IT portfolios contain a few workloads and ideas that can significantly improve a company's position in the market. Most cloud adoption efforts focus on the migration and modernization of existing workloads. Cloud innovation, however, can provide the greatest business value. Innovation related to cloud adoption unlocks new technical skills and expanded business capabilities.

In the Cloud Adoption Framework Innovate methodology, you focus on understanding customer needs and rapidly building innovations that shape how your customers interact with your products. This article illustrates an approach to delivering on the value of a minimum viable product (MVP).

To prepare for this phase of the cloud adoption lifecycle, the framework suggests the following cloud innovation exercises:

 Expand table

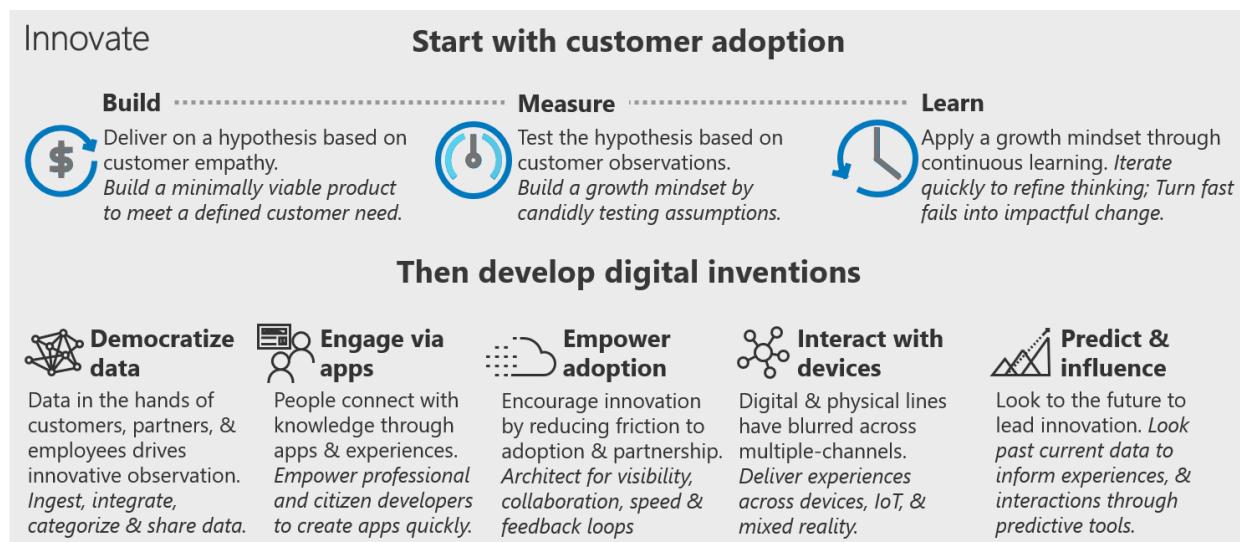
<p>1 Create hypothesis with business value consensus: Before you decide on technical solutions, identify how new innovations can drive business value and come up with a hypothesis about customer needs.</p> <p>2 Build your first MVP: Once you have a hypothesis with enough value potential to integrate it into your application, start the build process. Development sprints should be as quick as possible. Quick sprints let teams quickly verify or reject a hypothesis, or fine tune how required functionality integrates with the application.</p> <p>3 Measure & Learn from your MVP: You want to verify the accuracy of your hypothesis as soon as possible. A minimum viable product (MVP) is a preliminary version of the new feature that offers enough functionality to gather feedback and confirm if you're moving in the right direction.</p> <p>4 Expand digital innovation: To refine your hypothesis using the innovation disciplines or the digital inventions that include:</p> <ul style="list-style-type: none">• Democratize data• Engage via applications• Empower adoption• Interact with devices• Predict and influence

Innovation summary

The following approach builds on existing lean methodologies. It's designed to help you create a cloud-focused conversation about customer adoption and a scientific model for creating business value. The approach also maps existing Azure services to manageable decision processes. This alignment can help you find the right technical options to address specific customer needs or hypotheses.

As you build your MVP, you should consider using the five disciplines of digital inventions, which include:

- Democratize data
- Engage via applications
- Empower adoption
- Interact with devices
- Predict and influence



This article series emphasizes the following aspects of this methodology:

- First, always start with customer adoption to generate feedback that builds customer partnerships through the build-measure-learn feedback loop.
- Second, examine approaches to developing digital inventions that prioritize adoption.

You don't need to adopt all the practices at once. These disciplines highlight several approaches to developing digital inventions while also letting you focus on adoption and customer empathy.

The following section describes the formula for innovation and the commitments it takes to achieve success with this approach.

Formula for innovation

Successful innovation isn't about some significant transformational event or an elusive, mythical process. Success in innovation is more of a balancing act, illustrated by a simple equation: **innovation = invention + adoption**.

Innovation happens at the intersection of invention and adoption. True innovation stems from slowly adjusting human experiences through new approaches, new processes, and new technologies. In this formula, invention means you create a new solution that meets a customer need. Conversely, adoption means you apply the new solution to shape human behaviors and interactions. Finding the right balance between invention and adoption requires iteration, data-driven decision making, constant learning, and a growth mindset. It also requires technologies that can keep pace with the countless opportunities to learn in today's digital society.

The cloud is often a great platform for invention or the technological aspects of innovation. Unfortunately, most great ideas fail during the hard work of adoption, rather than during the ideation or invention processes. To ensure success, development teams should always start with adoption as the test for innovation. So this methodology starts with adoption. To use this methodology, your team should agree to the following three commitments:

- Commitment to prioritize customers over technology
- Commitment to transparency
- Commitment to iteration

Cultural commitments

Adopting the Innovate methodology requires cultural commitments to effectively use the metrics outlined in this article. Before you change your approach to driving innovation, make sure the adoption and leadership teams are ready to make these important commitments.

Commitment to prioritize customers over technology

Every development team has a set of tools or technologies that they're most familiar with. It's wise to play to those strengths and use what you know. However, for innovation to be successful, teams must maintain a focus on customer needs and the hypothesis being tested. At times, this focus might not align with the capabilities of a

particular tool or architectural approach. To be successful in innovation, the development team must remain open-minded. During the invention process, focus technical decisions on the needs of the customer over the preferences of your team.

Commitment to transparency

To understand measurement in an innovation approach, you must first understand the commitment to transparency. Innovation can only thrive in an environment that adheres to a *growth mindset*. A cultural imperative to learn from experiences is at the root of a growth mindset. Successful innovation and continuous learning start with a commitment to transparency in measurement. It's a brave commitment for the cloud adoption team to make. However, that commitment is meaningless if it's not matched by a commitment to preserve transparency within the leadership and cloud strategy teams.

Transparency is important because measuring customer impact doesn't address the question of right or wrong. Nor are impact measurements indicative of the quality of work or the performance of the adoption team. Instead, they represent an opportunity to learn and better meet your customers' needs. Misuse of innovation metrics can stifle that culture. Eventually, such misuse leads to manipulation of metrics, which in turn causes long-term failure of the invention, the supporting staff, and ultimately the management structure who misused the data. Leaders and contributors alike should avoid using measurements for anything other than an opportunity to learn and improve the MVP solution.

Commitment to iteration

Only one promise rings true across all innovation cycles: you won't get it right on the first try. Measurement helps you understand what adjustments you should make to achieve the desired results. Changes that lead to favorable outcomes stem from iterations of the build-measure-learn process. The cloud adoption team and the cloud strategy team must commit to an iterative mindset before adopting a growth mindset or a build-measure-learn approach.

Next steps

Before building the next great invention, review the different approaches to [developing digital inventions](#) while keeping adoption and customer empathy in mind.

[Develop digital inventions](#)

Feedback

Was this page helpful?

 Yes

 No

Azure innovative solutions guide overview

Article • 12/01/2022

ⓘ Note

This guide of innovative solutions provides a starting point for innovation guidance in the Cloud Adoption Framework. It's also available in the [Azure Quickstart Center](#).

Before you start developing innovative solutions by using Azure services, you need to prepare your environment, which includes preparing to manage customer feedback loops. In this guide, we introduce features that help you engage customers, build innovative solutions, and drive adoption. For more information, best practices, and considerations related to preparing your cloud environment, see the [Cloud Adoption Framework innovate section](#).

In this innovation guide, you'll learn how to:

- ✓ **Manage customer feedback:** Set up tools and processes to manage the build-measure-learn feedback loop by using GitHub and Azure DevOps.
- ✓ **Democratize data:** Data alone might be enough to drive innovative solutions to your customers. Deploy common data options in Azure.
- ✓ **Engage via applications:** Some innovation requires an engaging experience. Use cloud-native application platforms to create engaging experiences.
- ✓ **Empower adoption:** Invention is great, but a plan to reduce friction is needed to empower and scale adoption. Deploy a foundation for CI/CD, DevOps, and other adoption enablers.
- ✓ **Interact through devices:** Create ambient experiences to bring your applications and data closer to the customers' point of need. IoT, mixed reality, and mobile experiences are easier with Azure.
- ✓ **Predict and influence:** Find patterns in data. Put those patterns to work to predict and influence customer behaviors by using Azure-based predictive analytics tools.

💡 Tip

For an interactive experience, view this guide in the Azure portal. Go to the [Azure Quickstart Center](#) in the Azure portal, select **Azure innovation guide**, and then follow the step-by-step instructions.

Next steps:

- Prepare for innovation with a shared repository and ideation management tools

Prepare for customer feedback

Article • 02/16/2023

User adoption, engagement, and retention are key to successful innovation. Why?

Building an innovative new solution isn't about giving users what they want or think they want. It's about the formulation of a hypothesis that can be tested and improved upon. That testing comes in two forms:

- **Quantitative (testing feedback):** This feedback measures the actions we hope to see.
- **Qualitative (customer feedback):** This feedback tells us what those metrics mean in the customer's voice.

Quantitative data is number-based using a quantifiable measurement process.

Quantitative feedback gives numeric insights into data, which is useful for gathering a large number of answers from customers quickly. Examples of quantitative feedback would be multiple-choice questions and numerical user engagement data. Qualitative feedback is more in depth to get a wider variety of answers and insights into customer thoughts or opinions. Examples of qualitative feedback would be a customer survey with open-ended questions. Both methods of customer feedback provide valuable insights to improve your company's products and services.

Before you integrate feedback loops, you need to have a shared repository for your solution. A centralized repo will provide a way to record and act on all the feedback coming in about your project. [GitHub ↗](#) is the home for open-source software. It's also one of the most commonly used platforms for hosting source code repositories for commercially developed applications. The article on [building GitHub repositories](#) can help you get started with your repo.

Each of the following tools in Azure integrates with (or is compatible with) projects hosted in GitHub:

Quantitative feedback for web apps

Application Insights is a monitoring tool that provides near-real-time quantitative feedback on the usage of your application. This feedback can help you test and validate your current hypothesis to shape the next feature or user story in your backlog.

Action

To view quantitative data on your applications:

1. Go to **Application Insights**.
 - If your application doesn't appear in the list, select **add** and follow the prompts to start configuring Application Insights.
 - If the desired application is in the list, select it.
2. The **overview** pane includes some statistics on the application. Select **application dashboard** to build a custom dashboard for data that's more relevant to your hypothesis.

To view the data about your applications, go to the [Azure portal](#).

Learn more

- [Set up Azure Monitor](#)
- [Get started with Azure Monitor Application Insights](#)
- [Build a telemetry dashboard](#)

Data democratization tools

Article • 02/16/2023

Data democratization is the ability to make digital information accessible to the average non-technical user of information systems, without having a gatekeeper or outside help to access the data. Democratizing data helps users gain unfettered access to important data without creating a bottleneck that impedes productivity.

One of the first steps in democratizing data is to enhance data discoverability. Cataloging and managing data sharing can help enterprises get the most value from their existing information assets. By democratizing a data catalog it makes data sources easy to discover and understand by the users who manage the data. Azure Data Catalog enables management inside an enterprise, whereas Azure Data Share enables management and sharing outside the enterprise.

Azure services that provide data processing, like Azure Time Series Insights and Stream Analytics, are other capabilities that customers and partners are successfully using for their innovation needs.

Catalog

Azure Data Catalog

Azure Data Catalog addresses the discovery challenges of data consumers and enables data producers who maintain information assets. It bridges the gap between IT and the business, allowing everyone to contribute their insights. You can store your data where you want it and connect with the tools you want to use. With Azure Data Catalog, you can control who can discover registered data assets. You can integrate into existing tools and processes by using open REST APIs.

- ✓ Register
- ✓ Search and annotate
- ✓ Connect and manage

[Go to the Azure Data Catalog documentation](#)

Engage customers through applications

Article • 02/16/2023

Build cloud-native applications to connect customers in new ways. Cloud-native applications are built from the ground up, optimized for cloud scale and performance. Cloud-native applications are based on microservices architecture, use managed services, and take advantage of continuous delivery to achieve reliability and faster time to market.

Innovation with applications includes both modernizing your existing applications that are hosted on-premises and building cloud-native applications by using containers or serverless technologies. Azure provides PaaS services like Azure App Service to help you easily modernize your existing web and API apps written in .NET, .NET Core, Java, Node.js, Ruby, Python, or PHP for deployment in Azure.

With an open-standard container model, building microservices or containerizing your existing applications and deploying them on Azure is simple when you use managed services like Azure Kubernetes Service, Azure Container Instances, and Web App for Containers. Serverless technologies like Azure Functions and Azure Logic Apps use a consumption model (pay for what you use) and help you focus on building your application rather than deploying and managing infrastructure.

Deliver value faster

One of the advantages of cloud-based solutions is the ability to gather feedback faster and start delivering value to your user. Whether that user is an external customer or a user in your own company, the faster you can get feedback on your applications, the better.

Azure App Service

Azure App Service provides a hosting environment for your applications that removes the burden of infrastructure management and OS patching. It provides automation of scale to meet the demands of your users while bound by limits that you define to keep costs in check.

Azure App Service provides first-class support for languages like ASP.NET, ASP.NET Core, Java, Ruby, Node.js, PHP, and Python. If you need to host another runtime stack, Web App for Containers lets you quickly and easily host a Docker container within App Service, so you can host your custom code stack in an environment that gets you out of the server business.

Action

To configure or monitor Azure App Service deployments:

1. Go to **App Services**.
2. Configure a new service: select **Add** and follow the prompts.
3. Manage existing services: select the desired application from the list of hosted applications.

Azure Cognitive Services

With Azure Cognitive Services, you can infuse advanced intelligence directly into your application through a set of APIs that let you take advantage of Microsoft-supported AI and machine learning algorithms.

Action

To configure or monitor Azure Cognitive Services deployments:

1. Go to **Cognitive Services**.
2. Configure a new service: select **Add** and follow the prompts.
3. Manage existing services: select the desired service from the list of hosted services.

Azure Bot Service

Azure Bot Service extends your standard application by adding a natural bot interface that uses AI and machine learning to create a new way to interact with your customers.

Action

To configure or monitor Azure Bot Service deployments:

1. Go to **Bot Services**.
2. Configure a new service: select **Add** and follow the prompts.
3. Manage existing services: select the desired bot from the list of hosted services.

Azure DevOps

During your innovation journey, you'll eventually find yourself on the path to DevOps. Microsoft has long had an on-premises product known as Team Foundation Server (TFS). During our own innovation journey, Microsoft developed Azure DevOps, a cloud-based service that provides build and release tools supporting many languages and destinations for your releases. For more information, see [Azure DevOps](#).

Visual Studio App Center

As mobile apps continue to grow in popularity, the need for a platform that can provide automated testing on real devices of various configurations grows. Visual Studio App Center not only provides a place where you can test your cloud-native applications across iOS, Android, Windows, and macOS, it also provides a monitoring platform that can use Azure Application Insights to analyze your telemetry quickly and easily. For more information, see [Visual Studio App Center](#).

Visual Studio App Center also provides a notification service that lets you use a single call to send notifications to your application across platforms without having to contact each notification service individually.

Learn more

- [App Service overview](#)
- [Web App for Containers: Run a custom container](#)
- [Introduction to Azure Functions](#)
- [Azure for .NET and .NET Core developers](#)
- [Azure SDK for Python documentation](#)
- [Azure for Java cloud developers](#)
- [Create a PHP web app in Azure](#)
- [Azure SDK for JavaScript documentation](#)
- [Azure SDK for Go documentation](#)
- [DevOps solutions ↗](#)

Empower cloud adoption

Article • 12/01/2022

You know that innovation and digital transformation is critical to business success. Digital transformation is the adoption of cloud-based and digital technologies to replace old systems and create better customer experiences, better efficiency, business insights, and greater innovation from the data. You don't accomplish innovation solely through the introduction of new technologies. You need to focus on supporting the people who catalyze change and create the new value that you seek. Developers are at the center of digital transformation, and to empower them to achieve more, you need to accelerate developer velocity. To unleash the creative energy of developer teams, you need to help them build productively, foster global and secure collaboration, and remove barriers so they can scale innovation.

Generate value

- ✓ In every industry, each organization is trying to do one thing: drive constant value generation.
- ✓ The focus on innovation is essentially a process to help your organization find new ways to generate value.
- ✓ Perhaps the biggest mistake organizations make is trying to create new value by introducing new technologies.
- ✓ Sometimes the attitude is, "If we just use more technology, we'll see things improve." But innovation is first and foremost a people story.
- ✓ Innovation is about the combination of people and technology.

Organizations that successfully innovate toward digital transformation see vision, strategy, culture, unique potential, and capabilities as the foundational elements. They then turn to technology with a specific purpose in mind. Every company is becoming a software company. The hiring of software engineers is growing at a faster rate outside the tech industry than inside, according to LinkedIn data.

Innovation is accomplished when organizations support their people to create the value they seek. One group of those people, developers, is a catalyst for innovation. They play an increasingly vital role in value creation and growth across every industry. They're the builders of our era, writing the world's code and sitting at the heart of innovation. Innovative organizations build a culture that empowers developers to achieve more.

Developer productivity

Developer velocity

Empowering developers to invent means accelerating developer velocity, enabling them to create more, innovate more, and solve more problems. Developer velocity is the underpinning of each organization's tech intensity. Developer velocity isn't just about speed. It's also about unleashing developer ingenuity, turning your developers' ideas into software with speed and agility so that innovative solutions can be built. The differentiated Azure solution is uniquely positioned to unleash innovation and cloud adoption in your organization.

Build productively

There are several areas of opportunity where Azure can help you build productively:

- ✓ Ensure developers become and stay proficient in their domain by helping them advance their knowledge.
- ✓ Hone the right skills by giving them the right tools.

One of the best ways to improve your developers' skills is by giving them tools they know and love. Azure tools meet developers where they are today and introduce them to new digital transformation technologies in the context of the code they're writing. With the Azure commitment to open-source software and support for all languages and frameworks in Azure tools, your developers can build how they want and deploy where you want.

Azure DevOps provides best-in-class tools for every developer. Azure developer and digital transformation services infuse modern development practices and emerging trends into our tools. With the Azure platform, developers have access to the latest technologies and a cutting-edge toolchain that supports the way they work.

- ✓ AI-assisted development tools
- ✓ Integrated tools and cloud
- ✓ Remote development and pair programming

Go to [Azure DevOps documentation](#)

Interact through connected devices

Article • 02/16/2023

Innovate through intermittently connected devices and perceptive edge devices. Orchestrate millions of such devices, acquire and process limitless data, and take advantage of a growing number of multisensory, multidevice experiences. For devices at the edge of your network, Azure provides a framework for building immersive and effective business solutions. With ubiquitous computing, enabled by Azure combined with AI technology, you can build every type of intelligent application and system you can envision.

Ubiquitous computing is the processing of information that connects devices and processors to have constant availability, so that computing and processing are made to appear anytime and everywhere needed, using any connected device or perceptive edge device. Examples of ubiquitous computing include any system that sends information to another system to complete a task seamlessly, like a fitness watch that alerts that there is an incoming call from a cell phone and allows completion of the call through the watch, or systems that learn and adjust such as a thermostat or smart speakers.

Azure customers employ a continually expanding set of connected systems and devices that gather and analyze data (close to their users, the data, or both), with complete device management. Users get real-time insights and experiences, delivered by highly responsive and contextually aware applications. By moving parts of the workload to the edge, these connected devices can spend less time sending messages to the cloud and react more quickly to spatial events.

- ✓ Industrial assets
- ✓ Microsoft HoloLens 2
- ✓ Azure Sphere
- ✓ Azure Kinect DK
- ✓ Drones
- ✓ Azure SQL Edge
- ✓ IoT plug and play

Global scale IoT service

Architect solutions that exercise bidirectional communication with IoT devices at billions scale. Use out-of-box, device-to-cloud telemetry data to understand the state of your devices and define message routes to other Azure services just through configuration. By taking advantage of cloud-to-device messages, you can reliably send commands and notifications to your connected devices and track

message delivery with acknowledgment receipts. And you'll automatically resend device messages as needed to accommodate intermittent connectivity.

Here are a few features you'll find:

- **Security-enhanced communication** channel for sending and receiving data from IoT devices.
- **Built-in device management** and provisioning to connect and manage IoT and edge devices at scale.
- **Full integration with Event Grid** and serverless compute, simplifying IoT application development.
- **Compatibility with Azure IoT Edge** for building hybrid IoT applications.

Learn more

- [Azure IoT Hub](#)
- [Azure IoT Hub Device Provisioning Service \(DPS\)](#)
- [Use our modern IoT Azure DevOps project to help with your work item management ↗](#)

Kubernetes in the Cloud Adoption Framework

Article • 12/01/2022

Review a prescriptive framework that includes the tools, programs, and content (best practices, configuration templates, and architecture guidance) to simplify adoption of Kubernetes and cloud-native practices at scale.

The list of required actions is categorized by persona to drive a successful deployment of applications on Kubernetes, from proof of concept to production, then scaling and optimization.

Get started

To prepare for this phase of the cloud adoption lifecycle, use the following exercises:

- [Application development and deployment](#): Examine patterns and practices of application development, configure continuous integration and continuous delivery (CI/CD) pipelines, and implement site reliability engineering (SRE) best practices.
- [Cluster design and operations](#): Identify for cluster configuration and network design. Ensure future scalability by automating infrastructure provisioning. Maintain high availability by planning for business continuity and disaster recovery.
- [Cluster and application security](#): Familiarize yourself with Kubernetes security essentials. Review the secure setup for clusters and application security guidance.
- [AKS landing zone accelerator](#): The AKS landing zone accelerator provides an architectural approach and reference implementation that enables effective workload and scenario operationalization of landing zones on Azure, at scale and aligned with the Azure roadmap and the Microsoft Cloud Adoption Framework for Azure.

Application development and deployment

Article • 03/14/2023

Examine patterns and practices of application development, configure Azure Pipelines, and implement site reliability engineering (SRE) best practices. SRE is a software engineering approach to application development and deployment, change management, monitoring, and emergency response.

Plan, train, and proof

Use the following checklist and application development resources to plan your application development and deployment. You should be able to answer these questions:

- ✓ Have you prepared your application development environment and setup workflow?
- ✓ How will you structure the project folder to support Kubernetes application development?
- ✓ Have you identified the state, configuration, and storage requirements of your application?

SRE checklist

- **Prepare your development environment.** Configure your environment with the tools to create containers and set up your development workflow.

For more information, see:

- [Working with Docker in Visual Studio Code](#)
- [Working with Kubernetes in Visual Studio Code](#)
- [Bridge to Kubernetes overview](#)

- **Containerize your application.** Familiarize yourself with the end-to-end Kubernetes development experience, including application scaffolding, inner-loop workflows, application management frameworks, CI/CD pipelines, log aggregation, monitoring, and application metrics.

To learn more, see:

- [Get up and running with Kubernetes \(e-book collection\)](#)
- [Containerize Your Applications with Kubernetes on Azure \(webinar\)](#)

- **Review common Kubernetes scenarios.** Kubernetes is often thought of as a platform for delivering microservices, but it's becoming a broader platform. For more information about common Kubernetes scenarios, such as batch analytics and workflow, see [Overview of common Kubernetes scenarios \(video\)](#).
- **Prepare your application for Kubernetes.** Prepare your application file system layout for Kubernetes and organize for weekly or daily releases. Learn how the Kubernetes deployment process enables reliable, zero-downtime upgrades.

For more information, see:

- [How Kubernetes deployments work \(video\)](#)
- [Develop and deploy applications on Kubernetes](#)

- **Manage application storage.** Understand the performance needs and access methods for pods so that you can provide the appropriate storage options. Plan for ways to back up and test the restore process for attached storage.

To learn more, see:

- [Basics of stateful applications in Kubernetes \(video\)](#)
- [State and data in Docker applications](#)
- [Storage options in Azure Kubernetes Service](#)

- **Manage application secrets.** Use a key vault to store and retrieve keys and credentials. Don't store credentials in your application code.

For more information, see:

- [How Kubernetes and configuration management works \(video\)](#)
- [Understand secrets management in Kubernetes \(video\)](#)
- [Use Azure Key Vault with Kubernetes](#)
- [Use Azure AD workload identity to authenticate and access Azure resources](#)

Deploy to production and apply best practices

As you prepare the application for production, use the following checklist. You should be able to answer these questions:

- ✓ Can you monitor all aspects of your application?
- ✓ Have you defined resource requirements for your application? How about scaling requirements?
- ✓ Can you deploy new versions of the application without affecting production systems?

SRE best practices checklist

- **Configure readiness and liveness health checks.** Kubernetes uses readiness and liveness checks to know when your application is ready to receive traffic and when it needs to be restarted. When you don't define checks, Kubernetes can't determine if your application is running. For more information, see [Liveness and readiness checks ↗](#).
- **Configure logging, application monitoring, and alerting.** Monitoring your containers is critical, especially when you run a production cluster, at scale, with multiple applications. The recommended logging method for containerized applications is to write to the standard output (`stdout`) and standard error (`stderr`) streams.

For more information, see:

- [Logging in Kubernetes ↗](#)
- [Get started with monitoring and alerting for Kubernetes \(video\) ↗](#)
- [Azure Monitor for containers](#)
- [Enable and review Kubernetes control plane logs in Azure Kubernetes Service \(AKS\)](#)
- [View Kubernetes logs, events, and pod metrics in real time](#)

- **Define resource requirements for the application.** A primary way to manage the compute resources within a Kubernetes cluster is to use pod requests and limits. These requests and limits tell the Kubernetes scheduler what compute resources to assign to a pod. For more information, see [Define pod resource requests and limits](#).
- **Configure application scaling requirements.** Kubernetes supports horizontal pod autoscaling to adjust the number of pods in a deployment depending on CPU utilization or other select metrics. To use the autoscaler, all containers in your pods must have CPU requests and limits defined. To learn more, see [Configure horizontal pod autoscaling](#).
- **Deploy applications by using an automated pipeline and DevOps.** The full automation of all steps between code commit to production deployment allows teams to focus on building code and removes the overhead and potential human error in manual steps. Deploying new code is quicker and less risky, which helps teams become more agile, more productive, and more confident about their running code.

To learn more, see:

- [Evolve your DevOps practices](#)

- [Setting up a Kubernetes build pipeline \(video\)](#)
- [Deployment Center for Azure Kubernetes Service](#)
- [GitHub Actions for deploying to Azure Kubernetes Service](#)
- [Tutorial: Deploy from GitHub to Azure Kubernetes Service using Jenkins](#)

Optimize and scale

Now that the application is in production, use the application deployment checklist to optimize your workflow and prepare your application and team to scale. You should be able to answer these questions:

- ✓ Are cross-cutting application concerns abstracted from your application?
- ✓ Are you able to maintain system and application reliability, while still iterating on new features and versions?

Application deployment checklist

- **Deploy an API gateway.** An API gateway serves as an entry point to microservices, decouples clients from your microservices, adds another layer of security, and decreases the complexity of your microservices by removing the burden of handling cross-cutting concerns. For more information, see [Use Azure API Management with microservices deployed in Azure Kubernetes Service](#).
- **Deploy a service mesh.** A service mesh provides capabilities to your workloads, like traffic management, resiliency, policy, security, strong identity, and observability. Your application is decoupled from these operational capabilities, and the service mesh moves them out of the application layer and down to the infrastructure layer.

For more information, see:

- [How service meshes work in Kubernetes \(video\)](#)
- Learn about [service meshes](#)
- [Use Open Service Mesh with Azure Kubernetes Service](#)
- [Use Istio with Azure Kubernetes Service](#)
- [Use Linkerd with Azure Kubernetes Service](#)
- [Use Consul with Azure Kubernetes Service](#)

- **Implement SRE practices.** SRE is a proven approach that maintains crucial system and application reliability and iterates at the speed that the marketplace demands.

To learn more, see:

- [Introduction to site reliability engineering \(SRE\)](#)

- DevOps at Microsoft: Game streaming SRE ↗

Cluster design and operations

Article • 12/01/2022

This article covers cluster configuration and network design. Learn how to future-proof scalability by automating infrastructure provisioning. Provisioning is the process of setting up the IT infrastructure that you want. Automated infrastructure provisioning supports a remote installation and sets up virtual environments. It also helps you maintain high availability by planning for business continuity and disaster recovery.

Plan, train, and proof

As you get started, the checklist and Kubernetes resources below will help you plan the cluster design. By the end of this section, you'll be able to answer these questions:

- ✓ Have you identified the networking design requirements for your cluster?
- ✓ Do you have services with varying requirements? How many node pools are you going to use?

Checklist:

- **Identify network design considerations.** Understand cluster network design considerations, compare network models, and choose the Kubernetes networking plug-in that fits your needs. For Azure Container Networking Interface (CNI) networking, consider the number of IP addresses required as a multiple of the maximum pods per node (default of 30) and number of nodes. Add one node required during upgrade. When choosing load balancer services, consider using an ingress controller when there are too many services to reduce the number of exposed endpoints. For Azure CNI, the service CIDR has to be unique across the virtual network and all connected virtual networks to ensure appropriate routing.

To learn more, see:

- [Kubenet and Azure CNI networking](#)
- [Use kubenet networking with your own IP address ranges in Azure Kubernetes Service \(AKS\)](#)
- [Configure Azure CNI networking in Azure Kubernetes Service \(AKS\)](#)
- [Secure network design for an AKS cluster ↗](#)
- **Create multiple node pools.** To support applications that have different compute or storage demands, you can optionally configure your cluster with multiple node pools. For example, use more node pools to provide GPUs for compute-intensive

applications or access to high-performance SSD storage. For more information, see [Create and manage multiple node pools for a cluster in Azure Kubernetes Service](#).

- **Decide on availability requirements.** A minimum of two pods behind Azure Kubernetes Service ensures high availability of your application if there is pod failures or restarts. Use three or more pods to handle load during pod failures and restarts. For the cluster configuration, a minimum of two nodes in an availability set or virtual machine scale set is required to meet the service-level agreement of 99.95%. Use at least three pods to ensure pod scheduling during node failures and reboots.

To provide a higher level of availability to your applications, clusters can be distributed across Availability Zones. These zones are physically separate datacenters within a given region. When the cluster components are distributed across multiple zones, your cluster can tolerate a failure in one of the zones. Your applications and management operations remain available even if an entire datacenter experiences an outage. For more information, see [Create an Azure Kubernetes Service \(AKS\) cluster that uses Availability Zones](#).

Go to production and apply infrastructure best practices

As you prepare the application for production, implement a minimum set of best practices. Use this checklist at this stage. By the end of this section, you'll be able to answer these questions:

- ✓ Are you able to confidently redeploy the cluster infrastructure?
- ✓ Have you applied resource quotas?

Checklist:

- **Automate cluster provisioning.** With infrastructure as code, you can automate infrastructure provisioning to provide more resiliency during disasters and gain agility to quickly redeploy the infrastructure as needed. For more information, see [Create a Kubernetes cluster with Azure Kubernetes Service using Terraform](#).
- **Plan for availability using pod disruption budgets.** To maintain the availability of applications, define pod disruption budgets (PDB) to ensure that a minimum number of pods are available in the cluster during hardware failures or cluster upgrades. To learn more, see [Plan for availability using pod disruption budgets](#).

- **Enforce resource quotas on namespaces.** Plan and apply resource quotas at the namespace level. Quotas can be set on compute resources, storage resources, and object count. For more information, see [Enforce resource quotas](#).

Optimize and scale

Once the application is in production, how can you optimize your workflow and prepare your application and team to scale? Use the optimization and scaling checklist to prepare. By the end of this section, you'll be able to answer these questions:

- ✓ Do you have a plan for business continuity and disaster recovery?
- ✓ Can your cluster scale to meet application demands?
- ✓ Are you able to monitor your cluster and application health and receive alerts?

Checklist:

- **Automatically scale a cluster to meet application demands.** To keep up with application demands, you may need to adjust the number of nodes that run your workloads automatically using the cluster autoscaler. For more information, see [Configure Kubernetes cluster autoscaler](#).
- **Plan for business continuity and disaster recovery.** Plan for multiregion deployment, create a storage migration plan, and enable geo-replication for container images. To learn more, see [Best practices for region deployments - Azure Container Registry geo-replication](#).
- **Configure monitoring and troubleshooting at scale.** Set up alerting and monitoring for applications in Kubernetes. Learn about the default configuration, how to integrate more advanced metrics, and how to add custom monitoring and alerting to operate your application.
 - [Get started with monitoring and alerting for Kubernetes \(video\)](#)
 - [Configure alerts using Azure Monitor for containers](#)
 - [Review diagnostic logs for master components](#)
 - [Azure Kubernetes Service \(AKS\) diagnostics](#)

Cluster and application security

Article • 12/01/2022

Familiarize yourself with Kubernetes security essentials and review the secure setup for clusters and application security guidance. Kubernetes security is important throughout the container lifecycle because of the distributed, dynamic nature of a Kubernetes cluster. Applications are only as secure as the weakest link in the chain of services that comprise the application's security.

Plan, train, and proof

As you get started, the security essentials checklist and Kubernetes security resources below will help you plan for cluster operations and application security. By the end of this section, you'll be able to answer these questions:

- ✓ Have you reviewed the security and threat model of Kubernetes clusters?
- ✓ Is your cluster enabled for Kubernetes role-based access control?

Security checklist:

- **Familiarize yourself with the security essentials white paper.** The primary goals of a secure Kubernetes environment are ensuring that the applications it runs are protected, that security issues can be identified and addressed quickly, and that future similar issues will be prevented. For more information, see [The Definitive Guide to Securing Kubernetes \(white paper\)](#).
- **Review the security hardening setup for the cluster nodes.** A security hardened host OS reduces the surface area of attack and allows deploying containers securely. To learn more, see [Security hardening in AKS virtual machine hosts](#).
- **Setup cluster Kubernetes role-based access control (Kubernetes RBAC).** This control mechanism lets you assign users, or groups of users, permission to do things like create or modify resources, or view logs from running application workloads.

For more information, see

- [Understand Kubernetes role-based access control \(Kubernetes RBAC\) \(video\)](#)
- [Integrate Azure AD with Azure Kubernetes Service](#)
- [Limit access to cluster configuration file](#)

Deploy to production and apply Kubernetes security best practices

As you prepare the application for production, implement a minimum set of best practices. Use this checklist at this stage. By the end of this section, you'll be able to answer these questions:

- ✓ Have you set up network security rules for ingress, egress, and intra-pod communication?
- ✓ Is your cluster set up to automatically apply node security updates?
- ✓ Are you running a security scanning solution for your cluster and container services?

Security checklist:

- **Control access to clusters using group membership.** Configure Kubernetes role-based access control (Kubernetes RBAC) to limit access to cluster resources based on user identity or group membership. For more information, see [Control access to cluster resources using Kubernetes RBAC and Azure AD identities](#).
- **Create a secrets management policy.** Securely deploy and manage sensitive information, such as passwords and certificates, using secrets management in Kubernetes. For more information, see [Understand secrets management in Kubernetes \(video\)](#).
- **Secure intra-pod network traffic with network policies.** Apply the principle of least privilege to control network traffic flow between pods in the cluster. For more information, see [Secure intra-pod traffic with network policies](#).
- **Restrict access to the API server using authorized IPs.** Improve cluster security and minimize attack surface by limiting access to the API server to a limited set of IP address ranges. For more information, see [Secure access to the API server](#).
- **Restrict cluster egress traffic.** Learn what ports and addresses to allow if you restrict egress traffic for the cluster. You can use Azure Firewall or a third-party firewall appliance to secure your egress traffic and define these required ports and addresses. To learn more, see [Control egress traffic for cluster nodes in AKS](#).
- **Secure traffic with Web Application Firewall (WAF).** Use Azure Application Gateway as an ingress controller for Kubernetes clusters. For more information, see [Configure Azure Application Gateway as an ingress controller](#).

- **Apply security and kernel updates to worker nodes.** Understand the AKS node update experience. To protect your clusters, security updates are automatically applied to Linux nodes in AKS. These updates include OS security fixes or kernel updates. Some of these updates require a node reboot to complete the process. To learn more, see [Use kured to automatically reboot nodes to apply updates](#).
- **Configure a container and cluster scanning solution.** Scan containers pushed into Azure Container Registry and gain deeper visibility to your cluster nodes, cloud traffic, and security controls.

For more information, see:

- [Azure Container Registry integration with Defender for Cloud](#)
- [Azure Kubernetes Service integration with Defender for Cloud](#)

Optimize and scale

Now that the application is in production, how can you optimize your workflow and prepare your application and team to scale? Use the optimization and scaling checklist to prepare. By the end of this section, you'll be able to answer this question:

- ✓ Can you enforce governance and cluster policies at scale?

Security checklist:

- **Enforce cluster governance policies.** Apply at-scale enforcements and safeguards on your clusters in a centralized, consistent manner. To learn more, see [Control deployments with Azure Policy](#).
- **Rotate cluster certificates periodically.** Kubernetes uses certificates for authentication with many of its components. You might want to periodically rotate those certificates for security or policy reasons. To learn more, see [Rotate certificates in Azure Kubernetes Service \(AKS\)](#).

Develop digital inventions in Azure

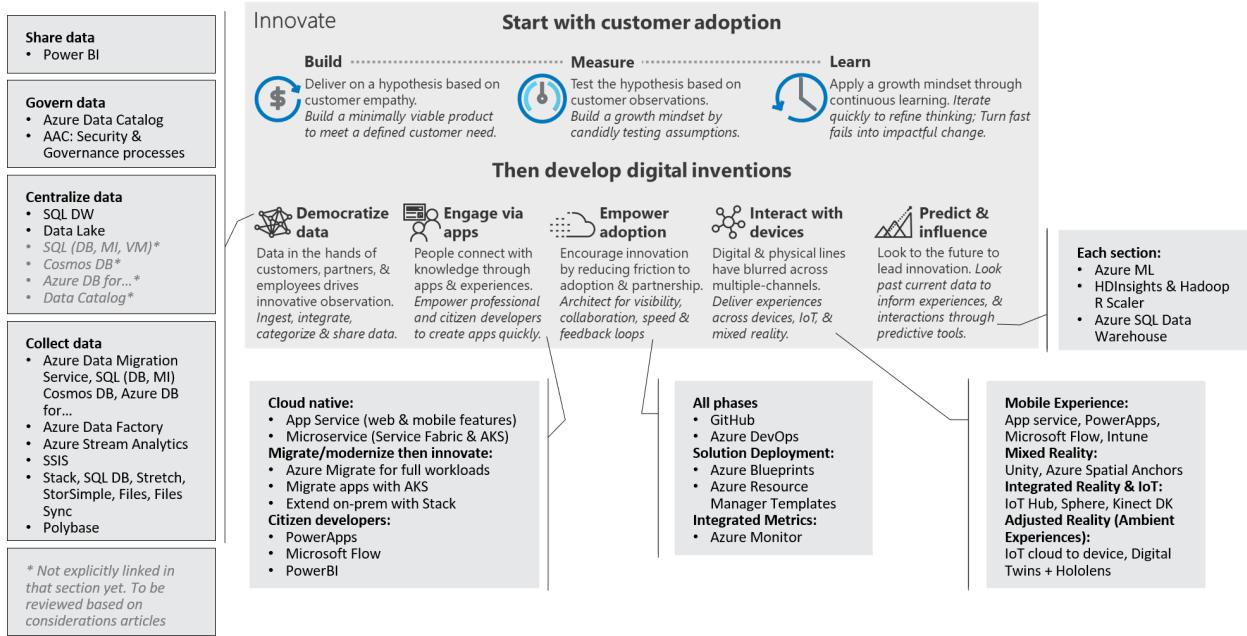
Article • 12/01/2022

Azure can help accelerate the development of each area of digital invention. This digital best practices section of the Cloud Adoption Framework builds on the [Innovate methodology](#). It shows how you can combine Azure services to create a toolchain for digital invention.

Alignment to the methodology

There are many combinations of cloud-based tools for digital invention and innovation within Azure. The following image shows an overview of how different digital invention tools align to each type of innovation.

Innovation Toolchain in Azure



Toolchain

The following article series demonstrates a few of the best practices and tools that closely align with the Innovate methodology. To test your hypothesis, start with the overview page of the type of digital invention you require. The overview page has the best practice guidance to act on so that you can [build with customer empathy](#).

Here are the types of digital invention in this article series:

- **Democratize data:** Tools for sharing data to solve information-related customer needs.

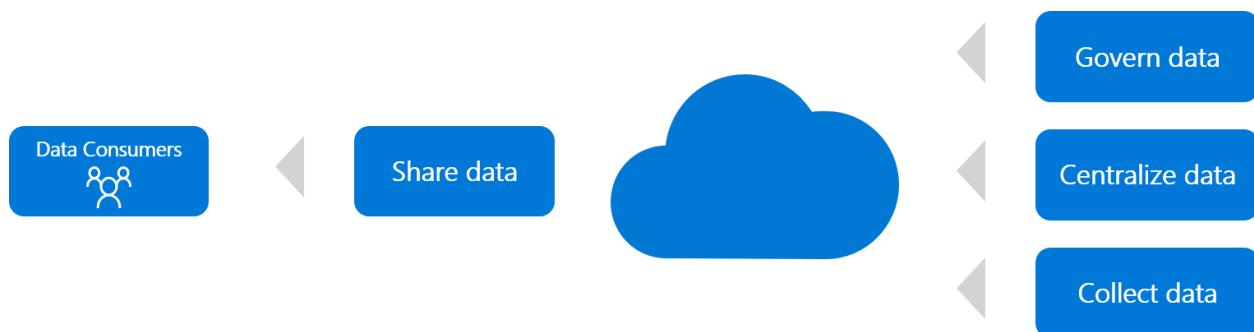
- **Engage via applications:** Tools to create applications that engage customers beyond raw data.
- **Empower adoption:** Tools to accelerate customer adoption through digital support for your build-measure-learn cycles.
- **Interact with devices:** Tools to create different levels of ambient experiences for your customers.
- **Predict and influence:** Tools for predictive analysis and integration of their output into applications.

Innovation tools to democratize data in Azure

Article • 09/12/2023

As the conceptual article on [democratizing data](#) describes, you can deliver many data collection innovations with little technical investment. Major innovations often require raw data. Democratizing data is about investing the least resources needed to engage your customers. The customers then use the data to take advantage of their existing knowledge.

Starting with data democratization is a quick way to test a hypothesis before expanding into broader, more costly digital inventions. As you refine more of the hypothesis and begin to adopt the inventions at scale, the following processes will help you prepare for operational support of the innovation.



Alignment to the methodology

This type of digital invention can be accelerated through each phase of the following processes, as shown in the preceding image. Technical guidance to accelerate digital invention is listed in the table of contents on the left side of this page. Those articles are grouped by phase to align guidance with the overall methodology.

- **Share collected data:** The first step of democratizing data is to share openly.
- **Govern data:** Ensure that sensitive data is secured, tracked, and governed before sharing.
- **Centralize data:** Sometimes you need to provide a centralized platform for data democratization, sharing, and governance.
- **Collect data:** Migration, integration, ingestion, and virtualization can each collect existing data to be centralized, governed, and shared.

In every iteration, cloud adoption teams should go only as deep into the stack as they require to put the focus on customer needs over architecture. Delaying technical spikes

in favor of customer needs accelerates the validation of your hypothesis.

All guidance maps to the four preceding processes. Guidance ranges from the highest customer effect to the highest technical effect. Across each process, you'll see guidance on ways Azure can accelerate your ability to [build with customer empathy](#).

Toolchain

In Azure, the following innovation tools are commonly used to accelerate digital invention across the preceding phases:

- [Power BI](#)
- [Azure Data Catalog](#)
- [Azure Synapse Analytics](#)
- [Azure Cosmos DB](#)
- [Azure Database for PostgreSQL](#)
- [Azure Database for MySQL](#)
- [Azure Database for MariaDB](#)
- [Azure Database for PostgreSQL hyperscale](#)
- [Azure Data Lake Storage](#)
- [Azure Database Migration Service](#)
- [Azure SQL Database, with or without Azure SQL Managed Instance](#)
- [Azure Data Factory](#)
- [Azure Stream Analytics](#)
- [SQL Server Integration Services](#)
- [Azure Stack](#)
- [SQL Server Stretch Database](#)
- [Azure StorSimple](#)
- [Azure Files](#)
- [Azure File Sync](#)
- [PolyBase](#)

As the invention approaches adoption at scale, the aspects of each solution require refinement and technical maturity. As that happens, more of these services are likely to be required. Use the table of contents on the left side of this page for Azure tools guidance relevant to your hypothesis-testing process.

Get started

Below you'll find articles to help you get started with each of the tools in this toolchain.

Note

The following links will leave the Cloud Adoption Framework, as they reference supporting content that's beyond the scope of CAF.

Share data with experts

- Quickly generate data insights
- Sharing data with coworkers and partners
- Embed reports in a website or portal
- Create new workspaces in Power BI

Govern data

- [Classify data \(CAF\)](#)
- [Secure data](#)
- [Annotate data with Azure Data Catalog](#)
- [Document data sources with Azure Data Catalog](#)

Centralize data

- [Create and query an Azure Synapse Analytics SQL pool](#)
- [Best practices for loading data for data warehousing](#)
- [Visualize warehouse data with Power BI](#)
- [Reference architecture for enterprise BI with Azure Synapse Analytics](#)
- [Manage enterprise big data with Azure Data Lake Storage](#)
- [What is a data lake?](#)

Collect data

- [Integrate cloud data sources with a SQL Analytics data warehouse](#)
- [Load on-premises data into Azure Synapse Analytics](#)
- [Integrate data - Azure Data Factory to OLAP](#)
- [Use Azure Stream Analytics with Azure Synapse Analytics](#)
- [Reference architecture for ingestion and analysis of new feeds](#)
- [Load data into Azure Synapse Analytics SQL pool](#)

Next steps

Learn about tools to create applications that engage customers beyond raw data.

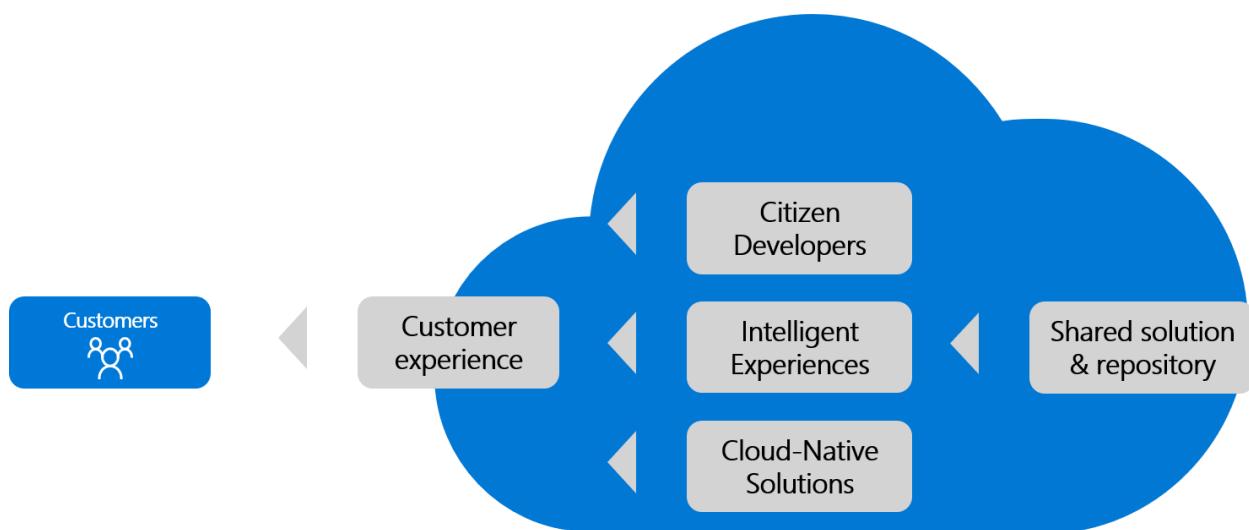
Engage via applications

Tools to engage with applications in Azure

Article • 09/12/2023

Build cloud-native applications to connect customers in new ways. Cloud-native applications are built from the ground up, optimized for cloud scale and performance. They are based on microservices architecture, use managed services, and take advantage of continuous delivery to achieve reliability and faster time to market.

As described in [Engage via applications](#), applications can be an important aspect of a minimal viable product (MVP) solution. For example, applications are often required for testing a hypothesis. This article helps you learn the application development tools that Azure provides to accelerate development of those applications.



Alignment with the Innovate methodology

You can accelerate this type of digital invention through each of the following approaches. Technical guidance for accelerating digital invention is listed in the table of contents on the left side of this page. Those articles are grouped by their approaches to aligning guidance with the overall Innovate methodology.

For this article, assume all inventions that result in an application stem from a shared solution as described in [Empower adoption](#). Also assume each application results in some type of customer experience for both internal and external customers.

Based on these assumptions, the following three paths are the most common for cloud adoption teams who are developing digital inventions:

- **Low-code application platform:** Empower business subject matter experts to create apps and automate business processes with visual tools that don't require professional developer skills.
- **Intelligent experiences:** Create modern experiences by using cloud platforms to drive rapid deployment and short feedback loops. Expand on web applications to infuse intelligence or even integrate bots.
- **Cloud-native:** Build a new invention that naturally takes advantage of cloud capabilities.

Each path results in advantages and disadvantages that are both short-term and long-term. When the cloud governance team, the cloud operations team, and the cloud center of excellence team are ready to support every approach, you can accelerate adoption with minimal effect on sustainable business operations.

Toolchain

Depending on the path that the cloud adoption team takes, Azure provides application development services and tools to accelerate the team's ability to build with customer empathy in mind. The following list of Azure offerings is grouped based on the preceding decision paths. These offerings include:

- Azure App Service
- Azure Kubernetes Service (AKS)
- Azure Migrate
- Azure Stack
- Power Apps
- Power Automate
- Power BI

Get started

Below you'll find articles to help you get started with each of the tools in this toolchain.

Note

The following links will leave the Cloud Adoption Framework, as they reference supporting content that's beyond the scope of CAF.

Low-code application platform

- Power Apps overview
- Creating applications in Power Apps
- Power Apps patterns
 - Patterns overview
 - Approval pattern
 - Asset management/resource booking pattern
 - Calculation/transformation pattern
 - Communication/announcement pattern
 - Inspection/audit pattern
 - Project management pattern
 - More patterns
- Plan a Power Apps project
- Real-world architecture examples
- Create a workflow with Power Automate
- Automate tasks with robotic process automation (RPA)
- Plan a Power Automate project

Intelligent experiences

- Modern web apps
 - Reference solutions for web apps
 - Create a .NET Core application with Azure SQL Database
 - Create a .NET application with Azure SQL Database
 - Create a PHP application with MySQL
 - Create a Node.js with MongoDB
- Infusing intelligence
 - Computer vision service
 - Translate text in real time
 - Understand sentiment using LUIS
 - Recognize speech from a microphone
 - Bing web search
 - Visual search from an image
 - Intelligent low-code apps
 - Chatbots
 - Choose the right chatbot solution
 - Create a bot with Composer
 - Create and deploy a Power Virtual Agents bot
 - Create a bot with Bot Framework SDK

Cloud-native applications

- Microservices architecture
 - [Design, build, and operate microservices in Azure](#)
 - [Reference architecture for microservices with Azure Kubernetes Service \(AKS\)](#)
 - [Reference architecture for serverless microservices](#)
- Containers
 - [What is Kubernetes? ↗](#)
 - [Prepare an application for Azure Kubernetes Service \(AKS\)](#)
 - [Create an Azure Red Hat OpenShift cluster](#)
- Spring Boot microservices
 - [Launch an Azure Spring Cloud application](#)
 - [Distributed tracing with Azure Spring Cloud](#)
 - [Bind an Azure Cosmos DB to your Azure Spring Cloud application](#)
- Event-driven applications
 - [Learning path for creating serverless applications](#)
 - [Azure Functions developer guide](#)
 - [Azure Serverless Computing Cookbook ↗](#)
 - [Azure Functions introduction](#)

Next steps

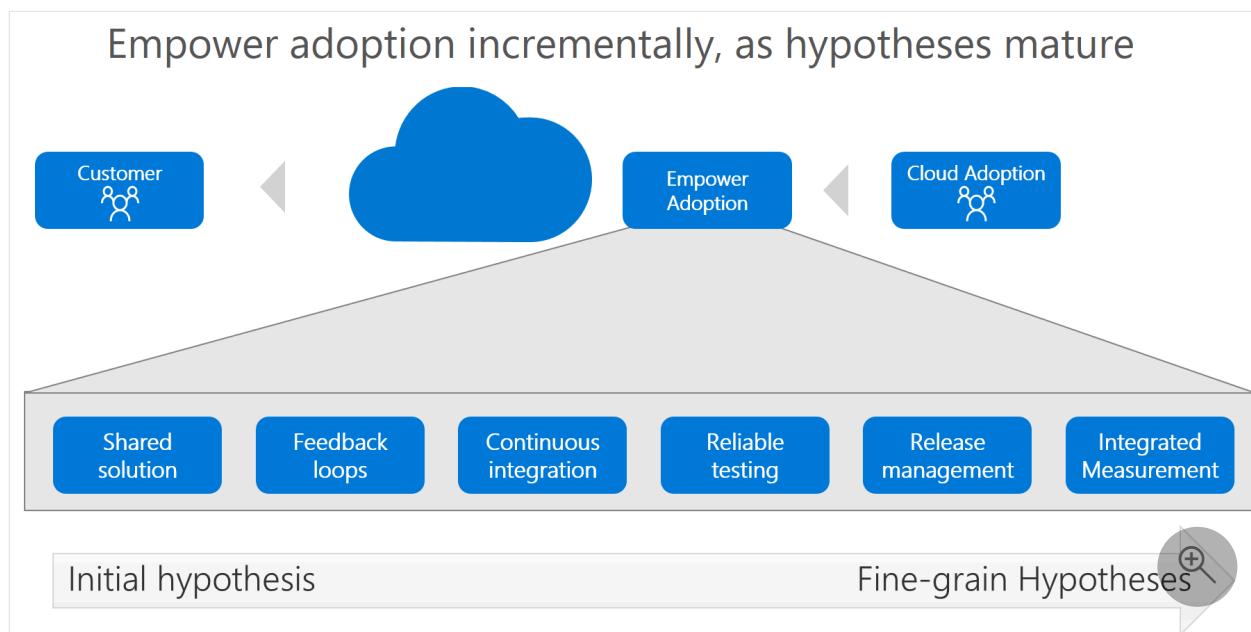
Learn about tools to accelerate customer adoption through digital support for your build-measure-learn cycles.

[Empower adoption](#)

Tools to empower adoption in Azure

Article • 05/14/2024

As described in [Empower adoption](#), building true innovation at scale requires an investment in removing friction that could slow adoption. In the early stages of testing a hypothesis, a solution is small. The investment in removing friction is likely small as well. As hypotheses prove true, the solution and the investment in empowering adoption grows. This article provides key continuous integration links to help you get started with each stage of the maturity model.



Alignment with the Innovate methodology

You can accelerate this type of digital invention through the following maturity levels. Technical guidance to accelerate digital invention is listed in the table of contents on the left side of this page. Those articles are grouped by maturity model level.

- **Shared solution:** Establish a centralized repository for all aspects of the solution.
- **Feedback loops:** Ensure feedback loops can be managed consistently throughout iterations.
- **Continuous integration:** Regularly build and consolidate a continuous integration and continuous delivery (CI/CD) solution.
- **Reliable testing:** Validate solution quality and expected changes to drive ensuring measurements.
- **Solution deployment:** Deploy a solution to allow a team to share changes with customers quickly.

- **Integrated measurement:** Add learning metrics to the feedback loop for clear analysis by the entire team.

Toolchain

For adoption teams that are mature professional development teams with many contributors, the Azure toolchain starts with GitHub and Azure DevOps.

You can expand this foundation to use other tool features as your need grows. The expanded foundation might involve tools like:

- Azure Policy
- Infrastructure as code (IaC) templates, such as Azure Resource Manager templates (ARM templates), Terraform templates, and Bicep templates
- Azure Monitor

The table of contents on the left side of this page lists guidance for each tool and aligns with the previously described maturity model.

Get started

Below you'll find articles to help you get started with each of the tools in this toolchain.

 **Note**

The following links will leave the Cloud Adoption Framework, as they reference supporting content that's beyond the scope of CAF.

Shared solution

- [Get started with a shared repository via GitHub ↗](#)
- [Get started with a shared backlog](#)
- [Synchronize Power Apps with Azure DevOps](#)

Feedback loops

- [Manage feedback with Azure DevOps](#)

Continuous integration

- Continuous integration with Azure Pipelines and GitHub ↗
- MLOps with Azure Machine Learning

Reliable testing

- Manage and track test plans

Solution deployment

- Continuous deployment with Azure Pipelines and GitHub ↗

Integrated metrics

- Monitor ASP.NET applications
- Monitor ASP.NET Core web applications
- Monitor Node.js applications
- Monitor mobile applications
- Monitor web applications
- Monitor VMs hosting traditional applications

Next steps

Learn about tools to accelerate customer adoption through digital support for your build-measure-learn cycles.

[Interact with devices](#)

Feedback

Was this page helpful?

 Yes

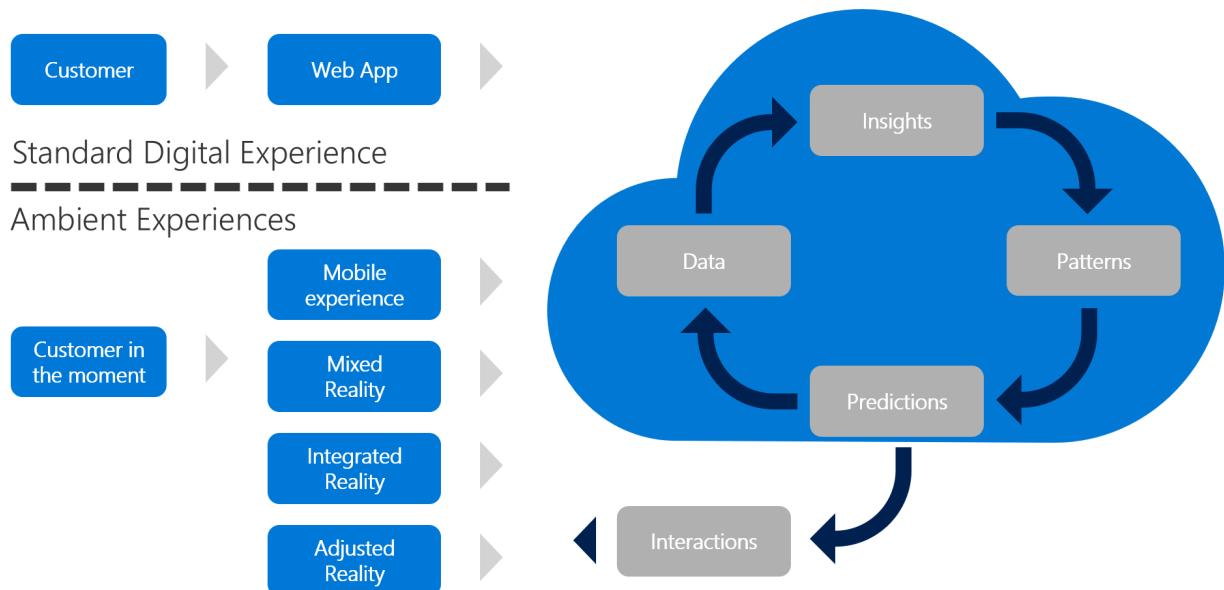
 No

Ambient experience tools to interact with devices in Azure

Article • 09/12/2023

As described in [Interacting with devices](#), the devices used to interact with a customer depend on how much ambient user experience is required to deliver the customer's need and empower adoption. Speed from the trigger that prompts the customer's need and your solution's ability to meet that need are determining factors in repeat usage. Ambient experiences help accelerate that response time and create a better experience for your customers by embedding your solution in the customers' immediate surroundings.

Interacting through devices & ambient experiences



Alignment to the methodology

This type of digital invention can be delivered through any of the following levels of ambient experience. The table of contents on the left side of this page lists technical guidance to accelerate digital invention. Those device interaction articles are grouped by level of ambient user experience to align with the methodology.

- **Mobile experience:** Mobile apps are commonly part of a customer's surroundings. In some scenarios, a mobile device might provide enough interactivity to make a solution ambient.
- **Mixed reality:** Sometimes a customer's natural surroundings must be altered through mixed reality. Engaging a customer within that mixed reality can provide a form of ambient user experience.

- **Integrated reality:** Moving closer to true ambience, integrated reality solutions focus on the use of a customer's physical device to integrate the solution into natural behaviors.
- **Adjusted reality:** When any of the preceding solutions use predictive analysis to provide an interaction with a customer within that customer's natural surroundings, that solution creates the highest form of ambient experience.

Toolchain

In Azure, you commonly use the following tools to accelerate digital invention across each level of ambient user experience solutions. These tools are grouped based on the amount of experience required to reduce complexity in aligning tools with those experiences.

Category	Tools
Mobile experiences	<ul style="list-style-type: none"> • Azure App Service • Power Apps • Power Automate • Intune
Mixed reality	<ul style="list-style-type: none"> • Unity • Azure Spatial Anchors • HoloLens
Integrated reality	<ul style="list-style-type: none"> • Azure IoT Hub • Azure Sphere • Azure Kinect DK
Adjusted reality	<ul style="list-style-type: none"> • IoT cloud to device • Azure Digital Twins + HoloLens

Get started

Below you'll find articles to help you get started with each of the tools in this toolchain.

Note

The following links will leave the Cloud Adoption Framework, as they reference supporting content that's beyond the scope of CAF.

Mobile experience

- Extend a legacy claims-processing application with a web and mobile experience
- Optimize reports to share data on a mobile app
- Extend Power Apps canvas application to a mobile experience
- Extend Power Automate to add a mobile experience
- **Secure mobile experiences**
- Protect mobile reports with face, touch, or passcodes
- Secure mobile task flows with Intune
- Secure data in common data services

Mixed reality

- Develop mixed reality experiences with Unity
- Quickstarts to add Azure Spatial Anchors to a mixed reality solution

Integrated reality and IoT

- Visualize sensor data with Azure IoT in Power BI
- Visualize sensor data with Azure IoT Hub in a web solution
- Securing an IoT solution
- Get started with Azure Sphere ↗
- Create a deployment with Azure Sphere
- Get started with Azure Kinect DK
- Build your first Azure Kinect DK application

Adjusted reality

- "Azure Digital Twins + HoloLens: Adjusting virtual reality" ↗
- Get started with Azure Digital Twins
- Monitor a building with Azure Digital Twins
- Azure IoT for cloud-to-device communications guide
- Azure IoT configuration for cloud-to-device communications

Next steps

Learn about tools for predictive analysis and integration of their output into applications.

Predict and influence

Innovation in the digital economy

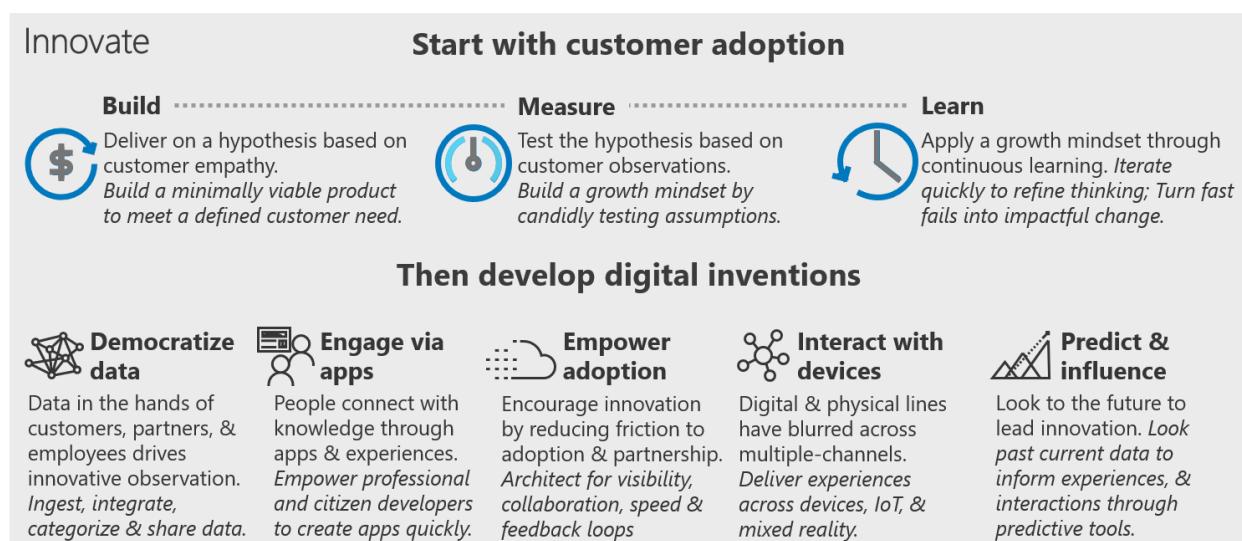
Article • 12/01/2022

The digital economy is an undeniable force in almost every industry. During the industrial revolution, gasoline, conveyor belts, and human ingenuity were key resources for promoting market innovation. Product quality, price, and logistics drove markets as companies sought to deliver better products to their customers more quickly. Today's digital economy shifts the way in which customers interact with corporations. The primary forms of capital and market differentiators have all shifted as a result. In the digital economy, customers are less concerned with logistics and more concerned with their overall experience of using a product. This shift arises from direct interaction with technology in our daily lives and from a realization of the value associated with those interactions.

In the Innovate methodology of the Cloud Adoption Framework, we'll focus on understanding customer needs and rapidly building innovations that shape how your customers interact with your products. We'll also illustrate an approach to delivering on the value of a minimum viable product (MVP). Finally, we'll map decisions common to innovation cycles to help you understand how the cloud can unlock innovation and create partnerships with your customers.

Innovate methodology

The simple methodology for cloud innovation within the Cloud Adoption Framework is illustrated in the following image. Subsequent articles in this section will show how to establish core innovation processes, approaches, and mechanisms for finding and driving innovation within your company.



This article series emphasizes the following aspects of this methodology:

- First, always start with customer adoption to generate feedback that builds customer partnerships through the build-measure-learn feedback loop.
- Second, examine approaches to developing digital inventions that prioritize adoption.

The following section describes what innovation is, the formula for innovation, and the commitments required for success with this approach.

Formula for innovation

Innovation is the development and application of ideas that improve the way things are done or what can be achieved.

Successful innovation is not a big-bang transformational event or an elusive magical unicorn. Success in innovation is more of a balancing act, illustrated by a simple equation: **innovation = invention + adoption**.

Innovation happens at the intersection of invention and adoption. True innovation stems from slowly adjusting human experiences through new approaches, new processes, and new technologies. In this formula, invention means creating a new solution that meets a customer need. Conversely, adoption means applying the new solution to shape human behaviors and interactions. Finding the right balance between invention and adoption requires iteration, data-driven decision making, constant learning, and a growth mindset. It also requires technologies that can keep pace with the countless opportunities to learn in today's digital economy.

The cloud is often a great platform for invention or the technological aspects of innovation. Unfortunately, most great ideas fail during the hard work of adoption, rather than during the creative or invention processes. To ensure success, development teams should always start with adoption as the test for innovation. To use this methodology, the team should agree on these three commitments:

- Commitment to prioritize customers over technology
- Commitment to transparency
- Commitment to iteration

Cultural commitments

Adopting the [Innovate methodology](#) requires some cultural commitments to effectively use the metrics outlined in this article. Before you change your approach to driving

innovation, make sure the adoption and leadership teams are ready to make these important commitments.

Commitment to prioritize customers over technology

Every development team has a set of tools or technologies that they're most familiar with. It's wise to play to those strengths and use what you know. However, for innovation to be successful, teams must maintain a focus on customer needs and the hypothesis being tested. At times, this focus may not align with the capabilities of a particular tool or architectural approach. To be successful in innovation, the development team must remain open-minded. During the invention process, focus technical decisions on the needs of the customer over the preferences of your team.

Commitment to transparency

To understand measurement in an innovation approach, you must first understand the commitment to transparency. Innovation can only thrive in an environment that adheres to a *growth mindset*. At the root of a growth mindset is a cultural imperative to learn from experiences. Successful innovation and continuous learning start with a commitment to transparency in measurement. This is a brave commitment for the cloud adoption team. However, that commitment is meaningless if it's not matched by a commitment to preserve transparency within the leadership and cloud strategy teams.

Transparency is important because measuring customer impact doesn't address the question of right or wrong. Nor are impact measurements indicative of the quality of work or the performance of the adoption team. Instead, they represent an opportunity to learn and better meet your customers' needs. Misuse of innovation metrics can stifle that culture. Eventually, such misuse will lead to manipulation of metrics, which in turn causes long-term failure of the invention, the supporting staff, and ultimately the management structure who misused the data. Leaders and contributors alike should avoid using measurements for anything other than an opportunity to learn and improve the MVP solution.

Commitment to iteration

Only one promise rings true across all innovation cycles: you won't get it right on the first try. Measurement helps you understand what adjustments you should make to achieve the desired results. Changes that lead to favorable outcomes stem from

iterations of the build-measure-learn process. The cloud adoption team and the cloud strategy team must commit to an iterative mindset before adopting a growth mindset or a build-measure-learn approach.

Next steps

Before building the next great invention, get started with customer adoption by understanding the [build-measure-learn feedback loop](#).

[Customer adoption with the build-measure-learn feedback loop](#)

Build consensus on the business value of innovation

Article • 07/28/2023

The first step in developing any new innovation is to identify how that innovation can drive business value. In this exercise, you answer a series of questions that highlight the importance of investing ample time when your organization defines business value.

What is business value?

Business value is an informal term that might vary from business to business. It's the net benefit to the customer and includes all forms of value that determine the long-term health and well-being of the company.

Qualifying questions to determine business value

Before you develop any solution for business innovation or business value (in the cloud or on-premises), validate your criteria for business value by answering the following questions:

1. What is the defined customer need that you want to address with this solution?
2. What opportunities will this solution create for your business?
3. Which business outcomes will this solution achieve?
4. Which of your company's motivations does this solution serve?

If the answers to all four questions are well documented, you might not need to complete the rest of this exercise.

Fortunately, you can easily test any documentation. Set up two short meetings to test both the documentation and your organization's internal alignment. Invite committed business stakeholders to one meeting and set up a separate meeting with the engaged development team. Ask the four questions to each group, and then compare the results.

Don't share the existing documentation with either team before the meeting. If true alignment exists, the members of each group should reference or even recite the guiding hypotheses.

 Tip

Don't facilitate the meeting. This test is to determine alignment. It's not an alignment creation exercise.

When you start the meeting, remind the attendees that the objective is to test directional alignment to existing agreements within the team. Establish a five-minute time limit for each question. Set a timer and close each question after five minutes, even if the attendees haven't agreed on an answer.

Account for the languages and interests of each group. If the test produces answers that are directionally aligned, consider this exercise a victory. You're ready to move on to solution development.

If one or two of the answers are directionally aligned, recognize that your hard work is paying off. You're already better aligned than most organizations. Future success is likely with minor continuing investment in alignment. Review each of the following sections for ideas that might help you build more alignment.

If either team fails to answer all four questions in 30 minutes, then alignment and the considerations in the following sections are likely to have a significant impact on this effort and others. Pay careful attention to each of the following sections.

Address the big picture first

The Cloud Adoption Framework follows a prescribed path through four phases: strategizing, planning, readiness, and adoption. Cloud and business innovation fits within the adoption phase of this process.

The answers to the third and fourth [qualifying questions](#) concern outcomes and motivations. When these answers are misaligned, your organization missed something during the strategizing phase of the cloud adoption lifecycle. One or more of the following scenarios are likely to be involved:

- **Alignment opportunity:** When business stakeholders can't agree on motivations and business outcomes related to a cloud and business innovation effort, it's a symptom of a larger challenge. The exercises in the [strategy methodology](#) can be useful in developing alignment among business stakeholders. We also recommend that the same stakeholders form a [cloud strategy team](#) that meets regularly.
- **Communication opportunity:** When the development team can't agree on motivations and business outcomes, the disagreement might be a symptom of strategic communication gaps. You can quickly resolve this problem by reviewing

the cloud strategy with the cloud adoption team. Several weeks after the review, the team should repeat the exercise of answering qualifying questions.

- **Prioritization opportunity:** A cloud strategy is essentially an executive-level hypothesis. The best cloud strategies are open to iteration and feedback.

If the cloud adoption team and the cloud strategy team understand the strategy but still can't align answers to the questions, priorities might be misaligned.

Organize a session with the teams. This session can help the efforts of both groups. The cloud adoption team starts by sharing its aligned answers to the qualifying questions. From there, a conversation between the cloud adoption team and cloud strategy team can highlight opportunities to better align priorities.

These big-picture opportunities often reveal ways to better align the innovative solution with the cloud strategy. This exercise often has one of these outcomes:

- These conversations can help your team improve your organization's cloud strategy and better represent important customer needs. Such a change can result in greater executive support for your team.
- These conversations might show that your cloud adoption team should invest in a different solution. In this case, consider migrating this solution before continuing to invest in innovation. Alternatively, these conversations might indicate that you adopt a citizen-developer approach to test the business value first. In either case, the conversations will help your team avoid making a large investment with limited business returns.

Address solution alignment

It's fairly common for the answers to the first and second questions to be misaligned.

During the early stages of ideation and development, customer need and business opportunity often get out of alignment. Many development teams find it challenging to achieve a balance between too much and too little definition. The Cloud Adoption Framework recommends lean approaches like build-measure-learn feedback loops to answer these questions.

The following list shows opportunities and approaches to create alignment:

- **Hypothesis opportunity:** Stakeholders and development teams might have too many expectations for a solution. Unrealistic expectations can be a sign that the hypothesis is too vague. Follow the guidance on [building with customer empathy](#) to construct a clearer hypothesis.

- **Build opportunity:** Teams might be misaligned because they disagree on the way to solve the customer need. Such disagreement typically indicates that [a premature technical spike](#) is delaying the team.

To keep the team focused on the customer, start the first iteration and build a small minimum viable product (MVP) to address part of the hypothesis. For more information, see [Develop digital inventions](#).

- **Training opportunity:** Either team can be misaligned because it needs deep technical requirements and extensive functional requirements. This need can lead to an opportunity for training in agile methodologies. When the team culture isn't ready for agile processes, you might find innovation and keeping pace with the market to be a challenge.

For training resources about DevOps and agile practices, see:

- [Get started with Azure DevOps](#)
- [Build applications with Azure DevOps](#)
- [Deploy applications with Azure DevOps](#)

By following the Cloud Adoption Framework Innovate methodology and the backlog management tools in each section of this article, you can help create solution alignment.

Next steps

After you've aligned your business value proposition and communicated it, you're ready to start building your solution.

[Return to the innovation exercises for next steps](#)

Create customer partnerships through the build-measure-learn feedback loop

Article • 07/03/2023

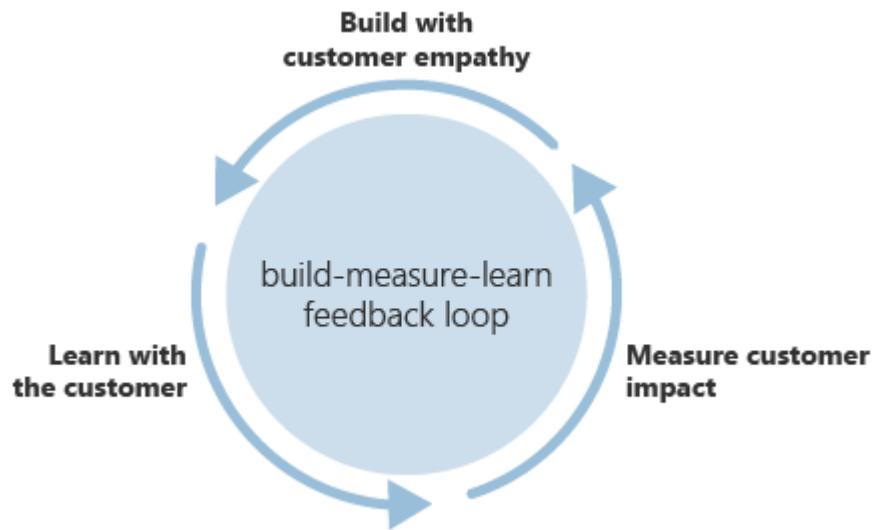
True innovation comes from the hard work of building solutions that demonstrate customer empathy, from measuring the impact of those changes on the customer, and from learning with the customer. Most importantly, it comes from feedback over multiple iterations.

If the past decade has taught us anything about innovation, it's that the old rules of business have changed. Large, wealthy incumbents no longer have an unbreakable hold on the market. The first or best players to market aren't always the winners. Having the best idea doesn't lead to market dominance. In a rapidly changing business climate, market leaders are the most agile.

Large or small, the companies that thrive in the digital economy as innovative leaders are those who listen to their customer base. That skill can be cultivated and managed. At the core of all good customer partnerships is a clear feedback loop. The process for building customer partnerships within the Cloud Adoption Framework is the build-measure-learn feedback loop.

What is the build-measure-learn feedback loop?

The build-measure-learn feedback loop is a description of the process for building empathy with your customers, measuring their reactions, and learning what adjustments to make that improve customer interactions. As described in [Innovation in the digital economy](#), innovation requires a balance between invention and adoption. Customer feedback and partnership drive adoption. By turning your customers into strong, loyal partners during innovation cycles, you can realize better products and gain quicker traction in the market.



This process for managing customer partnerships and integrating them into your innovation efforts includes three phases of development:

- Build with customer empathy
- Measure for customer impact
- Learn with customers

Each phase of the build-measure-learn feedback loop process helps you build better solutions with your customers.

Next steps

Learn how to [build with customer empathy](#) to begin your build-measure-learn cycle.

[Build with customer empathy](#)

Build with customer empathy

Article • 03/27/2023

"Necessity is the mother of invention." This saying captures the indelibility of the human spirit and our natural drive to invent. As explained in the Oxford English Dictionary, "When the need for something becomes imperative, you're forced to find ways of getting or achieving it." Few would deny these universal truths about invention. However, as the article [Innovation in the digital economy](#) explains, cloud innovation requires a balance between invention and adoption.

If we continue with the analogy, innovation comes from a more extended family. *Customer empathy is the proud parent of innovation.* To create a customer empathy solution, which drives innovation, requires a legitimate customer need that keeps them coming back to solve critical challenges. These solutions are based on what customers need rather than wants or whims. To find customers' true needs, start with empathy and a deep understanding of the customer experience. Empathy is an underdeveloped skill for many engineers, product managers, and even business leaders. Fortunately, the diverse interactions and rapid pace of the cloud architect role foster this skill.

How do you build empathy and why is customer empathy so important? Customer empathy helps you understand and share in the experience of the customer. From the first release of a minimum viable product (MVP), to the general availability of a market-grade solution, customer empathy helps you build better solutions. More importantly, empathy better positions a team to invent solutions that encourage adoption. In a digital economy, product teams who can most readily empathize with customer needs can build a brighter future with better tools that redefine and lead the market.

Define assumptions to build with customer empathy

Defining assumptions is a fundamental part of planning. The more you plan, the more clearly you can see your assumptions creeping into the foundation of a great idea. Assumptions are typically the product of self-empathy. In other words, *what would I want if I was in this position?* When you start with the build phase, it minimizes the period in which assumptions can invade a solution. This approach also accelerates the feedback loop with real customers, triggering earlier opportunities to learn and sharpen empathy.



Caution

Properly defining what to build can be tricky and requires some practice. If you build something too quickly, it might not reflect customer needs. If you spend too much time trying to understand initial customer needs and solution requirements, the market might meet them before you have a chance to build anything at all. In either scenario, the opportunity to learn can be significantly delayed or reduced. Sometimes the data can even be corrupted.

The most innovative solutions in history began with an intuitive belief. That gut feeling comes from both existing expertise and firsthand observation. Start with the build phase because it allows for a rapid test of your intuition. From there, you can cultivate deeper understanding and clearer degrees of empathy. At every iteration or release of a solution, balance comes from building MVPs that demonstrate customer empathy.

To steady this balancing act, the following two sections describe the concepts of how to build with empathy and define an MVP.

Define a customer-focused hypothesis

When you build with empathy, it means you create a solution based on defined hypotheses that illustrate a specific customer need. The following steps formulate a hypothesis that encourages building with empathy.

1. When you build with empathy, the customer is always the focus. This intention can take many shapes. You could reference a customer archetype, a specific persona, or even a picture of a customer in the midst of the problem you want to solve. And keep in mind that customers can be internal (employees or partners) or external (consumers or business customers). This definition is the first hypothesis for you to test: can we help this specific customer?
2. Understand the customer experience. Building with empathy means you can relate to the customer's experience and understand their challenges. This mindset indicates the next hypothesis to be tested: can we help this specific customer with this manageable challenge?
3. Define a clear solution to a single challenge. If you rely on expertise across people, processes, and subject matter experts, it can lead to a potential solution. The full hypothesis to be tested is then: can we help this specific customer with this manageable challenge through the proposed solution?
4. Arrive at a value statement. What long-term value do you hope to provide to these customers? The answer to this question creates your full hypothesis: how will these customers' lives be improved by using the proposed solution to address this manageable challenge?

This last step is the culmination of a customer empathy-driven hypothesis. It defines the audience, the problem, the solution, and the metric by which improvement is to be made, all of which center on the customer. During the measure and learn phases, you should test each hypothesis. Anticipate changes in the customer, the problem statement, or the solution as the team develops greater empathy for the addressable customer base.

Caution

The goal is to *build* with customer empathy, not to *plan* with it. It's all too easy to get stuck in endless cycles of planning and tweaking to hit upon the perfect customer empathy statement. Before you try to develop such a statement, review the following sections on defining and building an MVP.

After you prove the core assumptions, later iterations focus on growth tests in addition to empathy tests. After you build, test, and validate empathy, you begin to understand the addressable market at scale. You can more deeply understand your addressable market by expanding the standard hypothesis formula described previously. Then, based on available data, estimate the size of the total market (the number of potential customers).

From there, estimate the percentage of the total market that experiences a similar challenge and might therefore be interested in the solution. You then have your addressable market. The next hypothesis to test is: how will $x\%$ of customers' lives be improved by using the proposed solution to address this manageable challenge? A small sampling of customers reveals leading indicators that suggest a percentage effect on the pool of customers engaged.

Define a solution to test the hypothesis

During each iteration of a build-measure-learn feedback loop, your attempt to build with empathy is defined by an MVP.

An MVP is the smallest unit of effort (invention, engineering, application development, or data architecture) required to create enough of a solution to learn *with the customer*. The goal of every MVP is to test some or all of the prior hypotheses and to receive feedback directly from the customer. The output isn't a beautiful application with all the features required to change your industry. The desired output of each iteration is a learning opportunity, a chance to more deeply test a hypothesis.

Timeboxing is a standard way to make sure a product remains lean. For example, confirm that your development team thinks the solution can be created in a single iteration to allow for rapid testing. To better understand how to use velocity, iterations, and releases to define what minimal means, see [Planning velocity, iterations, release, and iteration paths](#).

Reduce complexity and delay technical spikes

The [disciplines of invention](#) described in [Innovate methodology](#) explore the functionality that's often required to deliver a mature innovation or scale-ready MVP solution. Use these disciplines as a long-term guide for feature inclusion. Likewise, use them as a cautionary guide during early testing of customer value and empathy in your solution.

Feature breadth and the different disciplines of invention can't all be created in a single iteration. It might take several releases for an MVP solution to include the complexity of multiple disciplines. Depending on the investment in development, there might be multiple parallel teams working within different disciplines to test multiple hypotheses. Although it's smart to maintain architectural alignment between those teams, it's unwise to try to build complex, integrated solutions until you can validate the value hypotheses.

You detect complexity the best in the frequency or volume of *technical spikes*. Technical spikes are efforts to create technical solutions that can't be easily tested with customers. When customer value and customer empathy are untested, technical spikes represent a risk to innovation, and should be minimized. For the types of mature tested solutions found in a migration effort, technical spikes can be common throughout adoption. But they delay the testing of hypotheses in innovation efforts and you should postpone them whenever possible.

A relentless simplification approach helps any MVP definition. This approach means you remove anything that doesn't aid your ability to validate the hypothesis. To minimize complexity, reduce the number of integrations and features that aren't required to test the hypothesis.

Build an MVP

At each iteration, an MVP solution can take many different shapes. The common requirement is only that the output allows for measurement and testing of the hypothesis. This simple requirement initiates the scientific process and lets the team build with empathy. To deliver this customer-first focus, an initial MVP might rely on only one of the [disciplines of invention](#).

In some cases, the fastest path to innovation means temporarily avoiding these disciplines entirely, until the cloud adoption team is confident that the hypothesis is accurately validated. From a technology company like Microsoft, this guidance might sound counterintuitive. But it emphasizes that customer needs, not a specific technology decision, are the highest priority in an MVP solution.

Typically, an MVP solution consists of an application or data solution with minimal features and limited polish. For organizations that have professional development expertise, this path is often the fastest one to learning and iteration. The following list includes several other approaches a team might take to build an MVP:

- A predictive algorithm that's wrong 99 percent of the time but that demonstrates specific desired outcomes.
- An IoT device that doesn't communicate securely at production scale but demonstrates the value of nearly real-time data within a process.
- An application built by a citizen developer to test a hypothesis or meet smaller-scale needs.
- A manual process that re-creates the benefits of the application to follow.
- A wireframe or video that's detailed enough to let the customer interact.

Developing an MVP shouldn't require massive amounts of development investment. Preferably, you constrain investment as much as possible to minimize the number of hypotheses being tested at one time. Then, in each iteration and with each release, you intentionally improve the solution toward your scale-ready solution that represents multiple disciplines of invention.

Accelerate MVP development

Time to market is crucial to the success of any innovation. Faster releases lead to faster learning. Faster learning leads to products that can scale more quickly. At times, traditional application development cycles can slow this process. More frequently, innovation is constrained by limits on available expertise. Budgets, headcount, and staff availability can all create limits to the number of new innovations a team can handle.

Staffing constraints and the desire to build with empathy spawned a rapidly growing trend toward citizen developers. These developers reduce risk and provide scale within an organization's professional development community. Citizen developers are subject matter experts in the customer experience, but they're not trained as engineers. These individuals use prototyping tools or lighter-weight development tools that might be frowned upon by professional developers. These business-aligned developers create MVP solutions and test theories. When aligned well, this process creates production

solutions that provide value but don't pass a sufficiently effective scale hypothesis. Teams can also use the solutions to validate a prototype before scale efforts begin.

Within any innovate plan, cloud adoption teams should diversify their portfolios to include citizen developer efforts. By scaling development efforts, you can form and test more hypotheses at a reduced investment. When you validate a hypothesis and identify an addressable market, professional developers can then harden and scale the solution by using modern development tools.

Final build gate: Customer pain

When customer empathy is strong, a clearly existing problem should be easy to identify. The customer's pain should be obvious. During build, the cloud adoption team is working on a solution to test a hypothesis based on a customer pain point. If the hypothesis is well-defined but the pain point isn't, the solution isn't truly based on customer empathy. In this scenario, build isn't the right starting point. Instead, invest first in building empathy and learning from real customers. The best approach for building empathy and validating pain is straightforward: listen to your customers. Invest time in meeting with and observing them until you can identify a pain point that occurs frequently. After you well understand the customer pain point, you're ready to test a hypothesized solution for addressing that pain.

When not to apply this approach

There are many legal, compliance, and industry requirements that might necessitate an alternate approach. This approach might not be suitable if public releases of a developing solution:

- Create risk to patent timing, intellectual property protection, or customer data leaks
- Violate established compliance requirements

When these perceived risks exist, consult legal counsel before adopting any guided approach to release management.

References

Some of the concepts in this article build on ideas discussed in [The Lean Startup](#) by Eric Ries.

Next steps

After you've built an MVP solution, you can measure the empathy value and scale value. Learn how to [measure for customer impact](#).

[Measure for customer impact](#)

How to measure for customer impact

Article • 12/01/2022

There are several ways to measure for customer impact. This article helps you define business metrics to validate hypotheses that arise out of an effort to [build with customer empathy](#).

Strategic metrics

The [Strategy methodology](#) examines [motivations](#) and [business outcomes](#). These practices provide a set of business metrics to test customer impact. When innovation is successful, results align with your strategic goals.

What are metrics in business? Business metrics are quantifiable metrics used to track and assess a specific business goal. Before establishing customer impact learning metrics, define strategic business metrics that you want this innovation to affect. Generally, those strategic metrics align with one or more of the following outcome areas:

- [Business agility](#)
- [Customer engagement](#)
- [Customer reach](#)
- [Financial impact](#)
- [Solution performance](#), in the case of operational innovation.

Document your business metrics and track their impact frequently. Don't expect results in any of these metrics to emerge for several iterations. For more information about setting and aligning expectations across the parties involved, see [Commitment to iteration](#).

Aside from motivation and business outcome metrics, this article focuses on learning metrics designed to guide transparent discovery and customer-focused iterations. For more information, see [Commitment to transparency](#).

Learning metrics

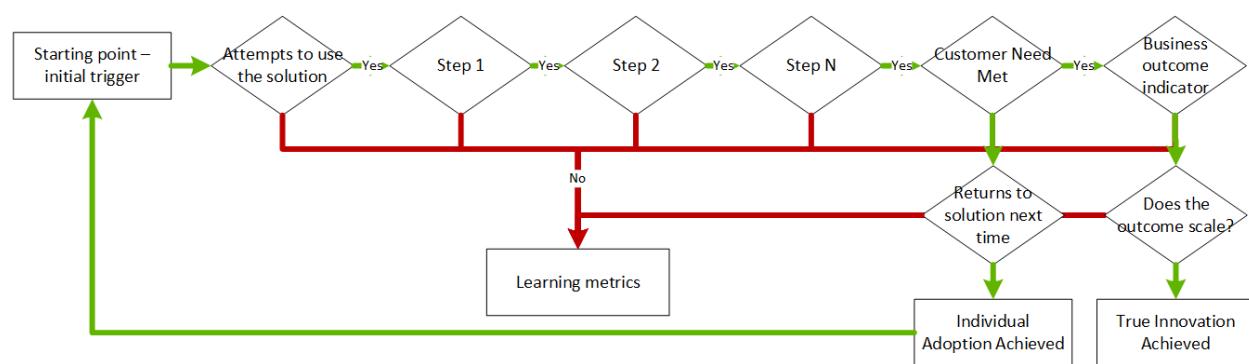
When you share the first version of a minimum viable product (MVP) with a customer, there won't be an impact on strategic metrics. Several iterations later, the team may still struggle to change behaviors enough to affect strategic metrics. During learning

processes, such as build-measure-learn cycles, consider adopting learning metrics. These metrics enhance tracking and learning opportunities.

Customer flow and learning metrics

If an MVP solution validates a customer-focused hypothesis, the solution drives some change in customer behavior. Those behavior changes across customer cohorts should improve business outcomes. Changing customer behavior is a process. Each step provides an opportunity to measure impact. The adoption team can keep learning along the way and build a better solution.

Learning about changes to customer behavior starts by mapping the flow that you hope to see from an MVP solution.



In most cases, a customer flow has an easily defined starting point and no more than two endpoints. Between the start and endpoints are various learning metrics to be used as measures in the feedback loop. Here are the steps to measure customer impact using the customer flow:

- **Starting point (initial trigger):** The starting point is the scenario that triggers the need for this solution. For a solution built with customer empathy, that initial trigger inspires a customer to try the MVP solution.
- **Solution steps:** Steps required to move the customer from the initial trigger to a successful outcome. Each step produces a learning metric based on a customer decision to move on to the next step.
- **Customer need met:** A solution that meets a customer need validates the hypothesis.
- **Individual adoption achieved:** If the customer returns to the solution the next time they find the trigger, individual adoption has been achieved.
- **Business outcome indicator:** When a customer behaves in a way that contributes to the defined business outcome, a business outcome indicator is observed.
- **True innovation achieved:** When *business outcome indicators* and *individual adoption* both occur at the desired scale, you've realized true innovation.

Each step of the customer flow generates learning metrics. After each iteration or release, a new version of the hypothesis is tested. At the same time, tweaks to the solution are tested to reflect adjustments in the hypothesis. When customers follow the prescribed path in any given step, a positive metric is recorded. When customers deviate from the prescribed path, a negative metric is recorded.

These alignment and deviation counters create learning metrics. Each should be recorded and tracked as the cloud adoption team progresses toward business outcomes and true innovation. In [Learn with customers](#), we'll discuss ways to apply these business metrics to learn and build better solutions.

Group and observe customer partners

The first measurement in defining learning metrics is the customer partner definition. Any customer who participates in innovation cycles qualifies as a customer partner. To accurately measure behavior, you should use a cohort model to define customer partners. In this model, customers are grouped to sharpen your understanding of their responses to changes in the MVP. These customer impact groups typically resemble the following groups:

- **Experiment or focus group:** Grouping customers based on their participation in a specific experiment designed to test changes over time.
- **Segment:** Grouping customers by the size of the company.
- **Vertical:** Grouping customers by the *industry vertical* they represent.
- **Individual demographics:** Grouping customers based on personal demographics like age and physical location.

These groupings help you validate learning metrics across cross-sections of those customers who choose to partner with you during your innovation efforts. All other metrics should be derived from definable customer groupings.

Next steps

As learning metrics accumulate, the team can begin to [learn with customers](#).

[Learn with customers](#)

Some of the concepts in this article build on topics first described in [The Lean Startup](#) ↗, written by Eric Ries.

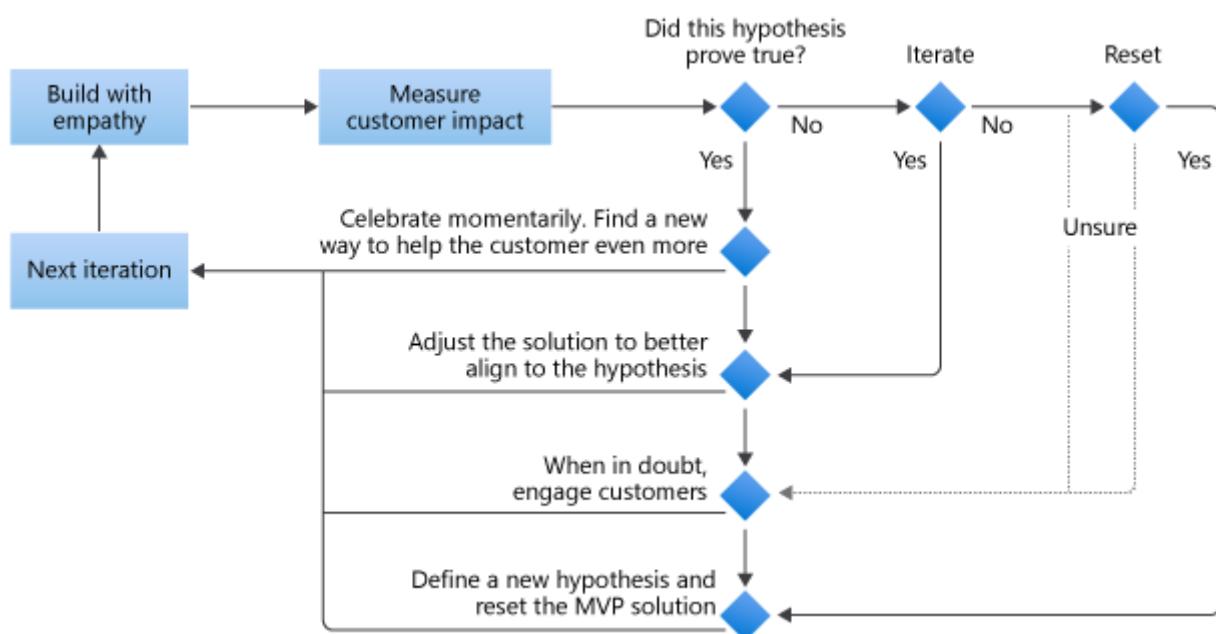
Learn with customers

Article • 12/01/2022

Our current customers represent our best resource for continuous learning. By partnering with us, customers help us [build with customer empathy](#) to find the best solution to their needs. They also help create a minimum viable product (MVP) solution by generating metrics from which we [measure customer impact](#). In this article, we'll describe how to drive innovation learning with and from our customer-partners.

Continuous learning

Continuous learning is the process of learning new skills and abilities continuously. This continuous learning can come from both formal training and taking on challenges. At the end of every iteration, we have an opportunity to learn from the build and measure cycles. The following image offers an overview of the continuous learning process flow.



Continuous learning is a method for responding to learning metrics and assessing their impact on customer needs. Here are the primary learning decisions to make at the end of each iteration:

- **Did the hypothesis prove true?** When the answer is yes, celebrate for a moment and then move on. There are always more things to learn, more hypotheses to test, and more ways to help the customer in your next iteration. When a hypothesis proves true, it's a good time to decide on a new feature that enhances the solution's utility for the customer.

- **Can you get closer to a validated hypothesis by iterating on the current solution?** The answer is usually yes. Learning metrics typically suggest points in the process that lead to customer deviation. Use these data to find the root of a failed hypothesis. At times, the metrics may also suggest a solution.
- **Is a reset of the hypothesis required?** Sometimes you learn that the hypothesis or underlying requirement was flawed. When you find a flaw, an iteration alone isn't necessarily the right answer. If a reset is required, rewrite the hypothesis and review the solution in light of the new hypothesis. The sooner this type of learning occurs, the easier it is to pivot. Early hypotheses should focus on testing the riskiest aspects of the solution to avoid pivots later in development.
- **Unsure?** The second most common response after "iterate" is "we're not sure". Embrace this response. This response represents an opportunity to engage the customer, foster customer empathy, and to look beyond the data.

The answers to these questions shape the iteration to follow. Companies that demonstrate an ability to apply continuous learning and boldly make the right decisions for their customers are more likely to emerge as leaders in their markets.

The practice of continuous learning requires a great deal of trial and error. It also requires some science and data-driven decision-making. Perhaps the most difficult part of adopting continuous learning concerns the cultural requirements. To effectively adopt continuous learning, your business culture must be open to a fail-first, customer-focused approach. The following section provides more details about this approach.

Growth mindset

Few could deny the radical transformation within Microsoft culture that's occurred over the last several years. This multifaceted transformation, led by Satya Nadella, has been hailed as a surprising business success story. At the heart of this story is the growth mindset. Here are a few key points of the growth mindset that inform learning with customers:

- **Customer first:** To design a hypothesis to improve the experience of customers, you have to meet real customers where they are. Don't just rely on metrics. Compare and analyze metrics based on firsthand observation of customer experiences.
- **Continuous learning:** Customer focus and customer empathy stem from a learn-it-all mindset. Strive for a learn-it-all, not know-it-all, mindset.
- **Beginner's mindset:** Demonstrate customer empathy by approaching every conversation with a beginner's mindset. Whether you're new to your field or a 30-year veteran, assume you know little, and you'll learn a lot.

- **Listen more:** Customers want to partner with you. An ego-driven need to be right blocks that partnership. To learn beyond the metrics, speak less and listen more.
- **Encourage others:** Don't just listen. Use the things you **do** say to encourage others. In every meeting, find ways to pull in diverse perspectives from those people who may not be quick to share.
- **Share the code:** If our obligation is to own a code base, we lose sight of the true power of innovation. Focus on owning and driving outcomes for your customers. Share your code, publicly with the world or privately within your company, to invite diverse perspectives into the solution and the code base.
- **Challenge what works:** Success doesn't mean you're demonstrating true customer empathy. Avoid having a fixed mindset and a bias toward doing what's worked before. Look for learning in positive and negative metrics by engaging your customers.
- **Be inclusive:** Work hard to invite diverse perspectives into the mix. There are many variables that can divide humans into segregated groups. Cultural norms, past behaviors, gender, religion, sexual preference, even physical abilities. True innovation comes when we challenge ourselves to see past our differences and consciously strive to include all customers, partners, and coworkers.

The build-measure-learn feedback loop

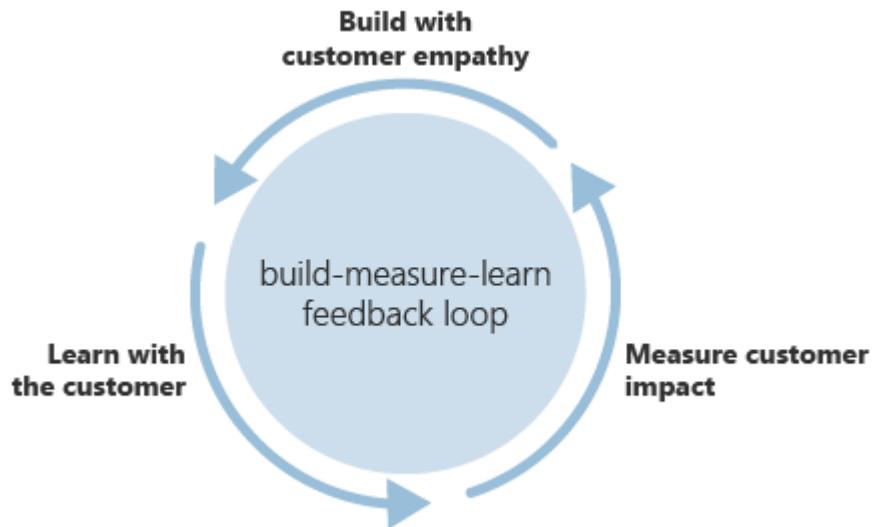
True innovation comes from the hard work of building solutions that demonstrate customer empathy, from measuring the impact of those changes on the customer, and from learning with the customer. Most importantly, it comes from feedback over multiple iterations.

If the past decade has taught us anything about innovation, it's that the old rules of business have changed. Large, wealthy incumbents no longer have an unbreakable hold on the market. The first or best players to market are not always the winners. Having the best idea doesn't lead to market dominance. In a rapidly changing business climate, market leaders are the most agile. Those who can adapt to changing conditions lead.

Large or small, the companies that thrive in the digital economy as innovative leaders are those organizations with the greatest ability to listen to their customer base. That skill can be cultivated and managed. At the core of all good partnerships is a clear feedback loop. The process for building customer partnerships within the Cloud Adoption Framework is the build-measure-learn feedback loop.

Innovation requires a balance between invention and adoption. Customer feedback and partnership drive adoption. By turning your customers into strong, loyal partners during

innovation cycles, you can realize better products and gain quicker traction in the market.



This process for managing customer partnerships and integrating them into your innovation efforts includes three phases of development:

- Build with customer empathy
- Measure for customer impact
- Learn with customers

Each phase of the process helps you build better solutions with your customers.

Next steps

As a next step to understanding this methodology, see [Common blockers and challenges to innovation](#) to prepare for the changes ahead.

Some of the concepts in this article build on topics first described in [The Lean Startup](#), written by Eric Ries.

Common technology adoption blockers and challenges to innovation

Article • 12/01/2022

As described in [Innovation in the digital economy](#), innovation requires a balance between invention and adoption. This article expands on the common cloud adoption challenges and blockers to innovation, as it aims to help you understand how this approach can add value during your innovation cycles.

Formula for innovation: **innovation = invention + adoption**

Knowing how to overcome innovation challenges takes some time to discover the right methods. This article delves into overcoming technology adoption challenges in the workplace.

Cloud technology adoption challenges

Although cloud technology advances have reduced some of the friction related to adoption, technology adoption is more people-centric than technology-centric. And unfortunately, the cloud can't fix people.

The following list describes some of the most common adoption challenges related to innovation. As you progress through the Innovate methodology, each of the challenges in the following sections are identified and addressed. Before you apply this methodology, evaluate your current innovation cycles to determine which are the most important challenges or blockers for you. Then, use the methodology to address or remove those blockers.

Types of external challenges

- **Time to market:** In a digital economy, time to market is one of the most crucial indicators of market domination. Surprisingly, time to market impact has little to do with positioning or early market share. Both of these factors are fickle and temporary. The time to market advantage comes from the simple truth that the more time your solution has on the market, the more time you have to learn, iterate, and improve. To shorten time to market and accelerate learning opportunities, focus on quick definition and rapid build of an effective minimum viable product.

- **Competitive challenges:** Dominant incumbents reduce opportunities to engage and learn from customers. Competitors also create external pressure to deliver more quickly. Build fast, but invest heavily in understanding the proper *measures*. Well-defined niches produce more actionable feedback measures and enhance your ability to partner and learn, resulting in better overall solutions.
- **Understand your customer:** Customer empathy starts with an understanding of the customer and customer base. One of the biggest challenges for innovators is the ability to rapidly categorize measurements and learning within the build-measure-learn cycle. It's important to understand your customer through the lenses of market segmentation, channels, and types of relationships. Throughout the build-measure-learn cycle, these data points help create empathy and shape the lessons learned.

Types of internal challenges

- **Choosing innovation candidates:** When investing in innovation, healthy companies spawn an endless supply of potential inventions. Many of these create compelling business cases that suggest high returns and generate enticing business justification spreadsheets. As described in the build article, *building with customer empathy* should be prioritized over invention that's based only on gain projections. If customer empathy isn't visible in the proposal, long-term adoption is unlikely.
- **Balancing the portfolio:** Most technology implementations don't focus on changing the market or improving the lives of customers. In the average IT department, more than 80% of workloads are maintained for basic process automation. With the ease of innovation, it's tempting to innovate and rearchitect those solutions. Most of the times, those workloads can experience similar or better returns by migrating or modernizing the solution, with no change to core business logic or data processes. Balance your portfolio to favor innovation strategies that can be *built* with clear empathy for the customer (internal or external). For all other workloads, follow a migrate path to financial returns.
- **Maintaining focus and protecting priorities:** When you've made a commitment to innovation, it's important to maintain your team's focus. During the first iteration of a build phase, it's relatively easy to keep a team excited about the possibilities of changing the future for your customers. However, that first MVP release is just the beginning. True innovation comes with each build-measure-learn cycle, by learning from the feedback loops to produce a better solution. As a leader in any innovation process, concentrate on keeping the team focused and on maintaining your innovation priorities through the subsequent, less-glamorous build iterations.

Invention challenges

Before the widespread adoption of the cloud, invention cycles that depended on information technology were laborious and time-consuming. Procurement and provisioning cycles frequently delayed the crucial first steps toward any new solutions. The cost of DevOps solutions and feedback loops delayed teams' abilities to collaborate on early stage ideation and invention. Costs related to developer environments and data platforms prevented anyone but highly trained professional developers from participating in the creation of new solutions.

The cloud has overcome many of these invention challenges by providing self-service automated provisioning, light-weight development and deployment tools, and opportunities for professional developers and citizen developers to cooperate in creating rapid solutions. Using the cloud for innovation dramatically reduces customer challenges and blockers to the invention side of the innovation equation.

Invention and innovation challenges in a digital economy

The invention challenges of today are different than challenges of the past. The endless potential of cloud technologies also produces more implementation options and deeper considerations about how those implementations might be used.

The Innovate methodology uses the following innovation disciplines to help align your implementation decisions with your invention and adoption goals:

- **Data platforms:** New sources and variations on data are available. Previously, much of this data couldn't be integrated into legacy or on-premises applications to create cost-effective solutions. Understanding the change you hope to drive in customers will inform your data platform decisions. Those decisions will be an extension of selected approaches to ingest, integrate, categorize, and share data. Microsoft refers to this decision-making process as the democratization of data.
- **Device interactions:** IoT, mobile, and augmented reality blur the lines between digital and physical, accelerating the digital economy. Understanding the real-world interactions surrounding customer behavior will drive decisions about device integration.
- **Applications:** Applications are no longer the exclusive domain of professional developers. Nor do they require traditional server-based approaches. Empowering professional developers, enabling business specialists to become citizen developers, and expanding compute options for API, micro-services, and PaaS solutions expand application interface options. Understanding the digital

experience required to shape customer behavior will improve your decision-making about application options.

- **Source code and deployment:** Collaboration between developers of all walks improves both quality and speed to market. Integration of feedback and a rapid response to learning shape market leaders. Commitment to the build, measure, and learn processes help accelerate tool adoption decisions.
- **Predictive solutions:** In a digital economy, it's seldom sufficient to just meet the current needs of your customers. Customers expect businesses to anticipate their next steps and predict their future needs. Continuous learning often evolves into prediction tooling. The complexity of customer needs and the availability of data will help define the best tools and approaches to predict and influence.

In a digital economy, the greatest challenge architects face is to clearly understand their customers' invention and adoption needs and to then determine the best cloud-based toolchain to deliver on those needs.

Next steps

With the knowledge you've gained about the build-measure-learn model and a growth mindset, you're ready to develop digital inventions within the [Innovate methodology](#).

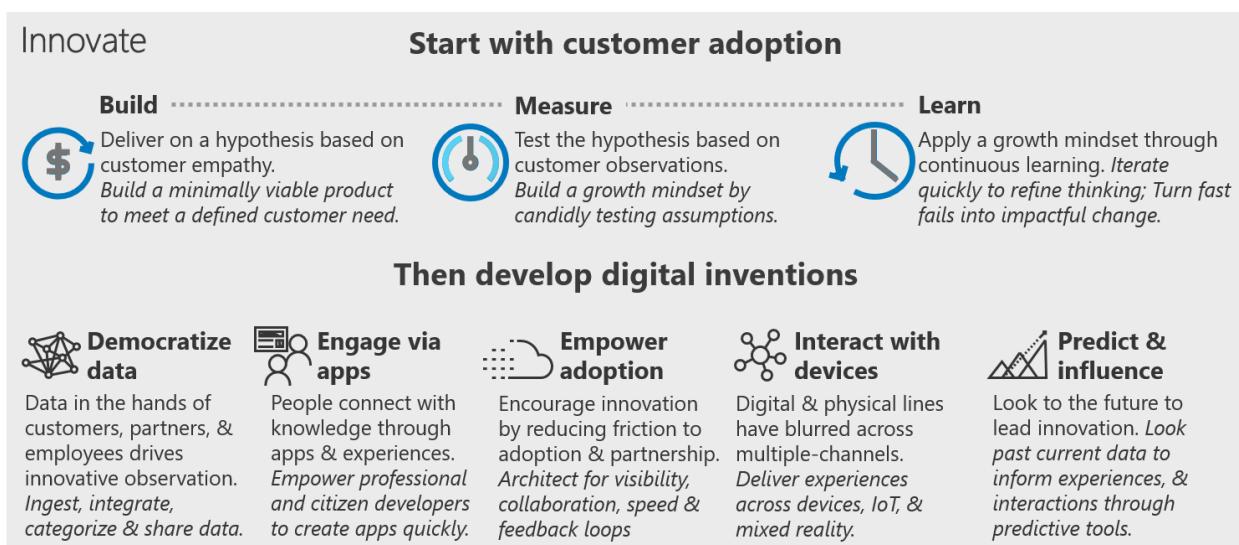
[Develop digital inventions](#)

Develop digital inventions

Article • 12/01/2022

What are digital inventions? As described in [Innovation in the digital economy](#), innovation requires a balance between invention and adoption. Digital inventions are the products of technological innovation that solve customer's needs and provide innovative solutions. Customer feedback and partnership are required to drive adoption, preferably using the build-measure-learn feedback loop. For more information, see [Create customer partnerships through the build-measure-learn feedback loop](#).

The types of innovation and disciplines described in the next section introduce a series of approaches to developing digital inventions while keeping adoption and customer empathy in mind. Each discipline is briefly described, along with links to each process.



Summary of each discipline of digital invention

There are different types of innovation for product development. Not every discipline is required to drive innovation for each specific case. By following the guidance in [Build with customer empathy](#), you'll test a hypothesis in every iteration. Defining the output of each iteration as a [minimum viable product \(MVP\)](#) should enable you to create innovation with the smallest number of disciplines.

- **Democratize data:** By getting data into the hands of customers, partners, and employees, you encourage innovative observation. Ingest, centralize, govern, and share data.
- **Engage via applications:** People connect with knowledge through applications and experiences. Empower professional and citizen developers to create applications quickly.

- **Empower adoption:** Encourage innovation by reducing friction to adoption and partnership. Architect for visibility, collaboration, speed, and feedback loops.
- **Interact with devices:** Digital and physical lines have blurred across multiple-channels. Deliver experiences across devices, IoT, and mixed reality.
- **Predict and influence:** Look to the future to lead innovation. Look past current data to inform experiences and interactions through predictive tools.

Next steps

Democratization of data is the first discipline of innovation to consider and evaluate.

Democratize data

Democratize data with digital invention

Article • 12/01/2022

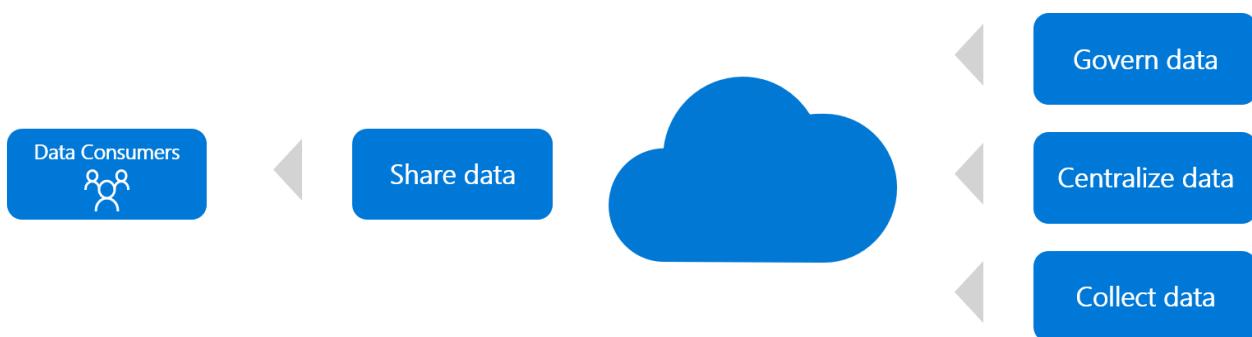
Coal, oil, and human potential were the three most consequential assets during the industrial revolution. These assets built companies, shifted markets, and ultimately changed nations. In the digital economy, there are three equally important assets for innovation: data, devices, and human potential. These assets holds great innovation potential. For any innovation effort in the modern era, data is the new oil.

In every company, there is data that can be used to find and meet customer needs. Unfortunately, the process of mining that data to drive innovation can be costly and time-consuming, so needs aren't discovered and solutions aren't created. Data democratization can solve this problem.

What is data democratization? It's the process of getting data into the right hands to drive innovation. This democratization process can take several forms, but they generally include solutions for ingested or integrated raw data, centralization of data, sharing data, and securing data. When data is democratized, experts around the company can use it to form and test hypotheses. In many cases, cloud adoption teams can [build with customer empathy](#) using only data, to rapidly meet customer needs.

Ways to democratize data

There are various ways to democratize data, but most include methods of collecting, centralizing, governing, and sharing the data. The following sections describe some of these methods. When you build a solution to a customer hypothesis, you should assess whether to democratize data, to what extent, and how to do it.



Share data

When you build with customer empathy, customer needs guide the solution. If the need is data, the solution enables the customer to interrogate, analyze, and report on the data directly, with no support from IT staff.

Many successful innovations start as a minimum viable product (MVP) that delivers data to the customer. An MVP is a version of the product that has just enough features to be usable by the customer. It shows the possible potential of the product in order to gather feedback from the customer. In this concierge model, an employee is the data consumer. That employee uses data to aid the customer. Each time the customer engages manual support, a hypothesis can be tested and validated. This approach is often a cost effective means of testing a customer-focused hypothesis before you invest heavily in integrated solutions.

The primary tools for sharing data directly with data consumers include self-service reporting or data embedded within other experiences, using tools like Power BI.

ⓘ Note

Before you share data, make sure you've read the following sections. Sharing data might require governance to provide protection for the data. Also, if the data spans multiple clouds it might require centralization. If data resides within applications, you must collect it in order to share it.

Govern data

Sharing data can quickly produce a minimum viable product to use in customer conversations. However, to turn that shared data into useful and actionable knowledge, more is generally required.

After a hypothesis has been validated through data sharing, the next phase of development is typically data governance.

Data governance is a broad topic that can require its own dedicated framework, a matter that's outside the scope of the [Cloud Adoption Framework](#).

There are several aspects of data governance to consider as soon as you validate the customer hypothesis. For example:

- **Is the shared data sensitive?** Data should be classified before being shared publicly to protect the interests of customers and the company.
- **If the data is sensitive, has it been secured?** Protection of sensitive data is a must for democratized data. The example workload discussed in [Securing data solutions](#) provides some references for securing data.
- **Is the data cataloged?** Identifying the nature of the shared data aids in long-term data management. Tools for documenting data, like [Azure Data Catalog](#) ↗, make

this process much easier in the cloud. Guidance regarding the [annotation of data](#) and the [documentation of data sources](#) can accelerate the process.

When democratization of data is important to a customer-focused hypothesis, make sure the governance of shared data is in the release plan. This protects customers, data consumers, and the company.

Centralize data

Data centralization leads to more meaningful reporting, ensures that the same data is available across the organization, and increases your ROI. When data is dispersed across an IT environment, opportunities to innovate can be extremely constrained, expensive, and time-consuming. The cloud provides new opportunities to centralize data. When centralization of multiple data sources is required to [build with customer empathy](#), the cloud can accelerate the testing of hypotheses.

⊗ Caution

Centralization of data represents a risk point in any innovation process. When data centralization is a technical spike, and not a source of customer value, we suggest that you delay centralization until the customer hypotheses have been validated.

When you centralize, you need an appropriate data store for the centralized data. It's a good practice to establish a data warehouse in the cloud. This scalable option provides a central location for all your data. This type of solution is available in online analytical processing (OLAP) or big data options.

The reference architectures for [OLAP](#) and [big data](#) solutions can help you choose the most appropriate centralization solution in Azure. If a hybrid solution is required, the reference architecture for [extending on-premises data](#) can also help accelerate solution development.

ⓘ Important

For some customer needs and solutions, a simple approach might be enough. The cloud architect should challenge the team to consider low-cost solutions to validate the customer hypothesis, especially during early development. This section on collecting data discusses scenarios that might suggest a different solution for your situation.

Collect data

The two primary forms of data collection are *integration* and *ingestion*.

Integration: Data that resides in an existing data store can be integrated into the centralized data store by using traditional data movement techniques. This is especially common for scenarios that involve multicloud data storage. These techniques involve extracting the data from the existing data store and then loading it into the central data store. At some point in this process, the data is typically transformed to be more usable and relevant in the central store.

Cloud-based tools have turned these techniques into pay-per-use tools, reducing the barrier to entry for data collection and centralization. Tools like [Azure Database Migration Service](#) and [Azure Data Factory](#) are two examples. The reference architecture for [Data Factory with an OLAP data store](#) is an example of one such solution.

Ingestion: Some data doesn't reside in an existing data store. When this transient data is a primary source of innovation, you'll want to consider alternative approaches. Transient data can be found in a variety of existing sources like applications, APIs, data streams, IoT devices, a blockchain, an application cache, in media content, or even in flat files.

You can integrate these various forms of data into a central data store on an OLAP or big data solution. However, for early iterations of the build-measure-learn cycle, an online transactional processing (OLTP) solution might be sufficient to validate a customer hypothesis. OLTP solutions aren't the best option for any reporting scenario. However, when you're [building with customer empathy](#), it's more important to focus on customer needs than on technical tooling decisions. After the customer hypothesis is validated at scale, a more suitable platform might be required. The reference architecture on [OLTP data stores](#) can help you determine which data store is most appropriate for your solution.

Virtualize: Integration and ingestion of data can sometimes slow innovation. When a solution for data virtualization is already available, it might represent a more reasonable approach. Ingestion and integration can both duplicate storage and development requirements, add data latency, increase attack surface area, trigger quality issues, and increase governance efforts. Data virtualization is a more contemporary alternative that leaves the original data in a single location and creates pass-through or cached queries of the source data.

SQL Server 2017 and Azure SQL Data Warehouse both support [PolyBase](#), which is the approach to data virtualization most commonly used in Azure.

Next steps

With a strategy for democratizing data in place, you'll next want to evaluate approaches to application development.

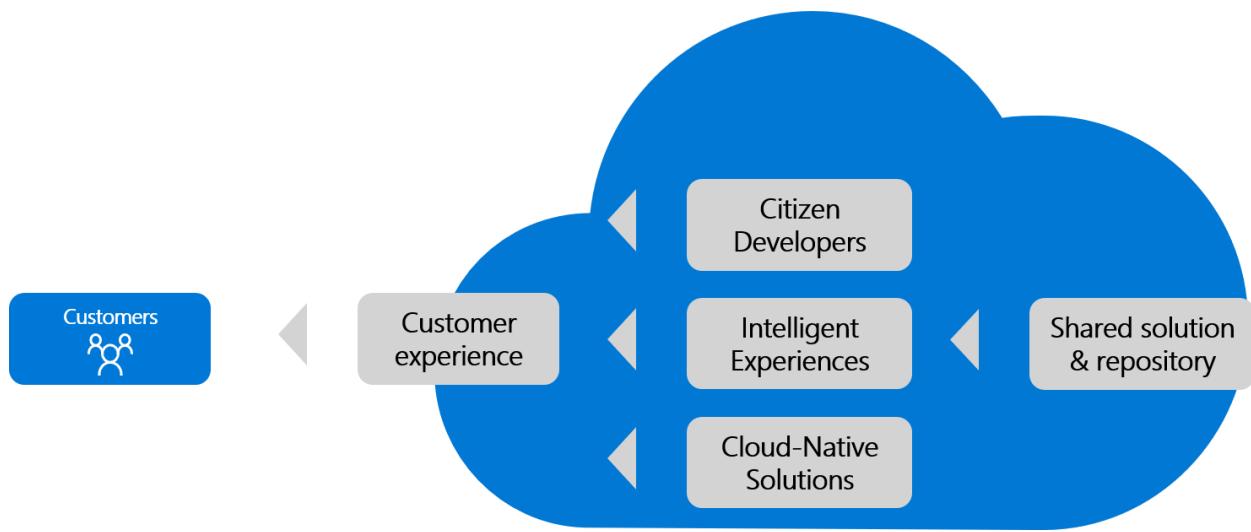
[Application development for innovative applications](#)

Application development for innovative applications

Article • 12/01/2022

As discussed in [Democratize data with digital invention](#), data fuels most innovations across the digital economy. Building on that analogy, applications are the fueling stations and infrastructure required to get that fuel into the right hands.

In some cases, data alone is enough to drive change and meet customer needs. More commonly though, solutions to customer needs require applications to shape the data and create an experience. Innovative applications engage and interact with the user, providing information and guidance. This article summarizes several principles that can help you find the right application development solution, based on the hypotheses to be validated.



Shared code

Teams that are quick to respond to customer feedback, market changes, and opportunities typically innovate best. The first principle of innovative applications is an element of the [growth mindset](#): "Share the code." Code sharing invites diverse perspectives and contributions, and spurs innovation. Therefore, application development should start with a shared code repository.

A widely adopted tool for managing code repositories is [GitHub](#) ↗, which allows you to create a shared code repository quickly. An alternative is [Microsoft Azure Repos](#) ↗, which is an [Azure DevOps](#) ↗ service that provides unlimited, cloud-hosted private repos for your project. For version control when you use Azure Repos, you can choose either Git, which is a distributed type, or Team Foundation Version Control (TFVC), which is

centralized. For more information about Azure Repos, Git, and TFVC, see the [Azure Repos documentation](#).

Citizen developers

Professional developers are important to innovation. When a hypothesis proves accurate at scale, they can stabilize the solution and prepare it for scale. Unfortunately, professional developers may be in short supply, and professional development can increase costs and slow innovation.

Citizen developers are users who create new business applications using development and runtime environments sanctioned by corporate IT. The use of citizen developers can help to scale development efforts and accelerate early hypothesis testing. This strategy is viable and effective when early hypotheses can be validated through tools like [Power Apps](#) for application interfaces, [AI Builder](#) for processes and predictions, [Power Automate](#) for workflows, and [Power BI](#) for data consumption.

ⓘ Note

When you rely on citizen developers to test hypotheses, it's advisable to also have professional developers to support, review, and guide the work. The professionals can help develop a robust design that accelerates returns on the innovation. By involving professional developers at the right time, you can realize cleaner transitions later.

Intelligent experiences

Intelligent experiences combine the speed and scale of modern web applications with the intelligence of cognitive services and bots. Individually, these technologies might be sufficient to meet your customers' needs. When properly combined, they broaden the spectrum of needs that can be met through a digital experience, while helping to contain application development costs.

Modern web apps

Modern web applications can be the fastest way to meet the needs of internal or external customers. The experiences they provide can engage customers quickly and allow for rapid evolution of the solution.

Adding intelligence

It gets easier all the time for professional and citizen developers to add machine learning and AI features to applications that help meet the needs of the customer and create an interactive experience. Some examples of these features are:

- Speech to text
- Text to speech
- Computer vision
- Visual search
- Predictive AI

Innovators should be alert to take advantage of such features to create an interactive and modern experience.

Bots

A bot is a conversational AI application that provides users an experience that is more like dealing with a person, and less like dealing with a conventional computer application. Users converse with a bot through text, interactive cards, and speech. A bot interaction can range from a quick question and answer, such as making a dinner reservation, to a sophisticated conversation that intelligently provides access to services.

Bots can do the same things as other types of software: read and write files, use databases and APIs, and handle regular computational tasks. What makes bots unique is their use of mechanisms generally reserved for human-to-human communication. Bots are a lot like modern web applications: they live on the internet and use APIs to send and receive messages. What's in a bot varies widely depending on what kind of bot it is. Modern bot software relies on a stack of technology and tools to deliver increasingly complex experiences on a variety of platforms. However, a simple bot could just receive a message and echo it back to the user with very little code involved.

Cloud-native solutions

Cloud-native architecture enables you to embrace rapid change, and run resilient and scalable applications more easily. Cloud-native applications are typically built using containers, [microservices](#), managed services, serverless functions, and event-based programming. Most commonly, cloud-native solutions use continuous delivery to achieve faster time to market.

A cloud-native solution allows centralized development teams to maintain control of the business logic without the need for monolithic, centralized solutions. It also creates an

anchor to drive consistency across the input of citizen developers and modern experiences. Finally, cloud-native solutions provide an innovation accelerator by freeing citizen and professional developers to innovate safely and with a minimum of blockers.

Innovate through existing solutions

Many customer hypotheses can best be delivered by a modernized version of an existing solution. This can happen when the current business logic comes close to meeting customer needs.

Most forms of modernization, including refactoring, are included in the [Migrate methodology](#) within the Cloud Adoption Framework. That methodology guides cloud adoption teams through the process of migrating a [digital estate](#) to the cloud. The [Azure migration guide](#) provides a streamlined approach to the same methodology, which is suitable for a small number of workloads or even a single application.

After a solution has been migrated and modernized, there are a variety of ways it can be used to create new, innovative application solutions to meet customer needs. For example, [citizen developers](#) could test hypotheses, or professional developers could create [intelligent experiences](#) or [cloud-native solutions](#).

Extend an existing solution

Extending a solution is one common form of modernization. This can be the fastest path to innovation when the following are true of the customer hypothesis:

- Existing business logic meets or comes close to customer needs.
 - An improved experience, not a new one, best meets the needs of customers.
 - The business logic required by the minimum viable product (MVP) solution has been centralized, usually via an [n-tier](#), web services, API, or [microservices](#) design.
- This approach consists of wrapping the existing solution within a new experience hosted in the cloud. In Azure, this solution would likely live in [Azure App Service](#) ↗.

Rebuild an existing solution

If an existing solution meets or comes close to meeting customer needs, but can't be easily extended, it may be necessary to refactor it. In this approach, the application is migrated to the cloud. After the application is migrated, parts of it are modified or duplicated, as web services or [microservices](#), which are deployed in parallel with the existing solution. The parallel service-based solution could be treated like an extended

solution. This solution would simply wrap the existing solution with a new experience hosted in the cloud. In Azure, this solution would likely live in Azure App Service.

⊗ Caution

Refactoring or rearchitecting solutions or centralizing business logic can quickly trigger a time-consuming **technical spike** instead of a source of customer value. This is a risk to innovation, especially early in hypothesis validation. With a bit of creativity in the design of a solution, there should be a path to MVP that doesn't require refactoring of existing solutions. It's wise to delay refactoring until the initial hypothesis can be validated at scale.

Operating model innovations

In addition to modern innovative approaches to application development, there have been notable innovations in application operations. These approaches have spawned many organizational movements. One of the most prominent is the [cloud center of excellence](#) operating model. When fully staffed and mature, business teams have the option to provide their own operational support for a solution.

The type of self-service operational management model found in a cloud center of excellence allows for tighter controls and faster iterations within the solution environment. These goals are accomplished by transferring operational control and accountability to the business team.

If you're trying to scale or meet global demand for an existing solution, this approach might be sufficient to validate a customer hypothesis. After a solution is migrated and slightly modernized, the business team can scale it to test a variety of hypotheses. These typically involve customer cohorts who are concerned with performance, global distribution, and other customer needs hindered by IT operations.

Reduce overhead and management

The more there is to maintain within an innovative application or solution, the slower that application or solution will iterate. This means you can accelerate innovation by reducing the impact of operations on available bandwidth.

To prepare for the many iterations required to deliver an innovative solution, it's important to think ahead. For example, minimize operational burdens early in the

process by favoring serverless options. In Azure, serverless application options could include [Azure App Service](#) or [containers](#).

In parallel, consider the serverless transaction data options in Azure that can also reduce overhead. The [Azure product catalog](#) provides database options that host data without the need for a full data platform.

Next steps

Depending on the hypothesis and solution, the principles in this article can aid in designing applications that meet MVP definitions and engage users. Up next are the principles for [empowering adoption](#), which offer ways to get the application and data into the hands of customers more quickly and efficiently.

[Empower adoption](#)

Empower adoption with digital invention

Article • 12/01/2022

The ultimate test of innovation is customer reaction to your invention. Did the hypothesis prove true? Do customers use the solution? Does it scale to meet the needs of the desired percentage of users? Most importantly, do they keep coming back? None of these questions can be asked until the minimum viable product (MVP) solution has been deployed.

In this article, we'll focus on empowering adoption with continuous integration and continuous deployment (CI/CD) pipeline tools. Continuous integration is the automating of code multiple times per day in order to have an updated single project. Continuous deployment is the automatic delivery of those functions throughout the day.

Reduce CI/CD friction that affects adoption

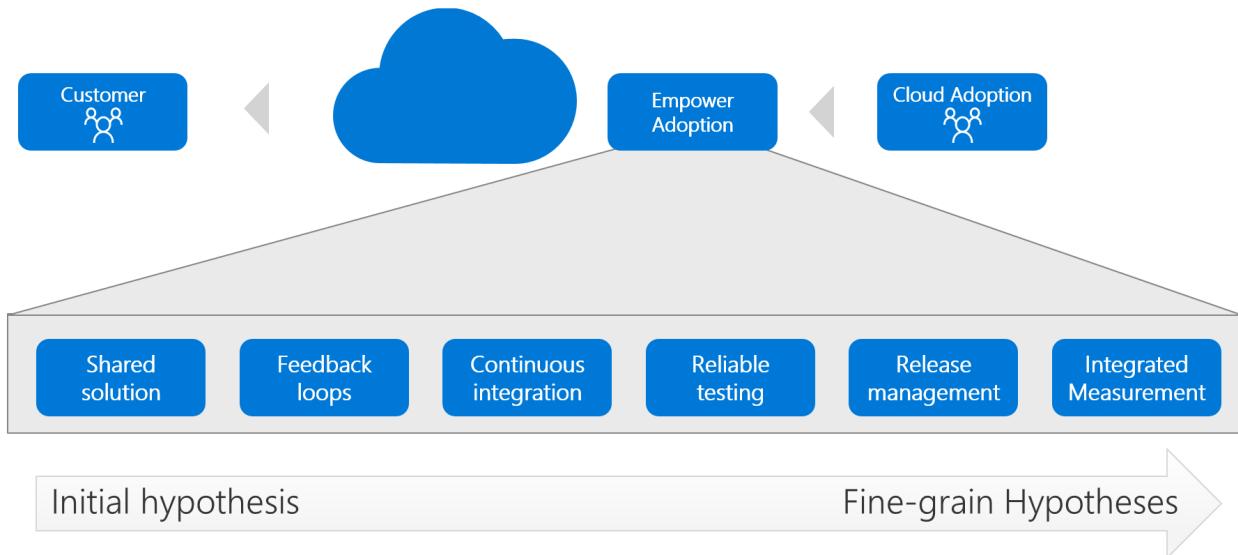
Some obstacles to adoption can be minimized through a combination of technology and processes. For readers with knowledge of CI/CD or DevOps processes, the following CI/CD pipeline processes will be familiar. This article establishes a starting point for cloud adoption teams that fuels innovation and feedback loops. This starting point might foster more robust CI/CD or DevOps approaches as the products and teams mature.

As described in [Measure for customer impact](#), positive validation of any hypothesis requires iteration and determination. This CI/CD article aims to minimize [technical spikes](#) that slow innovation, while making sure you keep best practices in place. Doing so will help the team design for future success while delivering on current customer needs.

Empower adoption and digital invention: The maturity model

The primary goal of the [Innovate methodology](#) is to build customer partnerships and accelerate feedback loops, which lead to market innovations. The following image and sections describe initial implementations that support this methodology.

Empower adoption incrementally, as hypotheses mature



- **Shared solution:** Establish a centralized repository for all aspects of the solution.
- **Feedback loops:** Make sure that feedback loops can be managed consistently through iterations.
- **Continuous integration:** Regularly build and consolidate the solution.
- **Reliable testing:** Validate solution quality and expected changes to ensure the reliability of your testing metrics.
- **Solution deployment:** Deploy solutions so that the team can quickly share changes with customers.
- **Integrated measurement:** Add learning metrics to the feedback loop for clear analysis by the full team.

To minimize technical spikes, assume that maturity will initially be low across these principles. Plan ahead by aligning to tools and processes that can scale as hypotheses become more fine-grained. In Azure, [GitHub](#) and [Azure DevOps](#) allow small teams to get started with little friction. These teams might grow to include thousands of developers who collaborate on scale solutions and test hundreds of customer hypotheses. The rest of this article illustrates the "plan big, start small" approach to empowering adoption across these principles.

Shared solution

As described in [Measure for customer impact](#), positive validation of any hypothesis requires iteration and determination. You'll experience far more failures than wins during any innovation cycle. This is expected. However, when a customer need, hypothesis, and solution align at scale, the world changes quickly.

When you're scaling digital invention and innovation, there's no more valuable tool than a shared code base for the solution. Unfortunately, there's no reliable way of predicting

which iteration or which MVP will yield the winning combination. That's why it's never too early to establish a shared code base or repository. This is the one [technical spike](#) that shouldn't be delayed. As the team iterates through various MVP solutions, a shared repo enables easy collaboration and accelerated development. When changes to the solution drag down learning metrics, version control lets you roll back to an earlier, more effective version of the solution.

The most widely adopted CI/CD tool for managing code repositories is [GitHub](#), which lets you create a shared code repository in just a few steps. Additionally, the [Azure Repos](#) feature of Azure DevOps can be used to create a [Git](#) or [TFVC](#) repository.

Feedback loops

Making the customer part of the solution is the key to building customer partnerships during innovation cycles. That's accomplished, in part, by [measuring customer impact](#). It requires conversations and direct testing with the customer. Both generate feedback that must be managed effectively.

Every point of feedback is a potential solution to the customer need. More importantly, every bit of direct customer feedback represents an opportunity to improve the partnership. If feedback makes it into an MVP solution, celebrate that with the customer. Even if some feedback isn't actionable, simply being transparent with the decision to deprioritize the feedback demonstrates a [growth mindset](#) and a focus on [continuous learning](#).

Azure DevOps includes ways to [request, provide, and manage feedback](#). These tools centralizes feedback so that the team can take action and provide follow-up in service of a transparent feedback loop.

Continuous integration

Continuous integration is the automating of code multiple times per day to have an updated single project. As adoptions scale and a hypothesis gets closer to true innovation at scale, the number of smaller hypotheses to be tested tends to grow rapidly. For accurate feedback loops and smooth adoption processes, it's important that those hypotheses are integrated and supportive of the primary hypothesis behind the innovation. This requires you to move quickly to innovate and grow, which requires multiple developers for testing variations of the core hypothesis. For later stage development efforts, you might even need multiple teams of developers, each building toward a shared solution. Continuous integration is the first step toward management of all the moving parts.

In continuous integration, code changes are frequently merged into the main branch. Automated build and test processes make sure that code in the main branch is always production quality. This ensures that developers are working together to develop shared solutions that provide accurate and reliable feedback loops.

Azure DevOps and [Azure Pipelines](#) provide continuous integration capabilities with just a few steps in GitHub or other repositories. For more information, see [What is continuous integration?](#) or try the [continuous integration hands-on lab ↗](#). Solution architectures are available that can accelerate creation of your [CI/CD pipelines via Azure DevOps ↗](#).

Reliable testing

Defects in any solution can create false positives or false negatives. Unexpected errors can easily lead to misinterpretation of user adoption metrics. They can also generate negative feedback from customers that doesn't accurately represent the test of your hypothesis.

During early iterations of an MVP solution, defects are expected. Early adopters might even find them endearing. In early releases, acceptance testing is typically nonexistent. However, one aspect of building with empathy concerns the validation of the need and hypothesis. Both can be completed through unit tests at a code level and manual acceptance tests before deployment. Together, these provide some means of reliability in testing. You should try to automate a well-defined series of build, unit, and acceptance tests. These will ensure reliable metrics related to finer tweaks to the hypothesis and the resulting solution.

The [Azure Test Plans](#) feature provides tooling to develop and operate test plans during manual or automated test execution.

Solution deployment

Perhaps the most meaningful aspect of empowering adoption is your ability to control the release of a solution to customers. By providing a self-service or automated pipeline for releasing a solution to customers, you'll accelerate the feedback loop. By allowing customers to quickly interact with changes in the solution, you invite them into the process. This approach also triggers quicker testing of hypotheses, reducing assumptions and potential rework.

There are several methods for solution deployment. The three most common are:

- **Continuous deployment** is the most advanced method, as it automatically deploys code changes into production. For mature teams that are testing mature hypotheses, continuous deployment can be extremely valuable.
- During early stages of development, **continuous delivery** might be more appropriate. In continuous delivery, any code changes are automatically deployed to a production-like environment. Developers, business decision-makers, and others on the team can use this environment to verify that their work is production-ready. You can also use this method to test a hypothesis with customers without affecting ongoing business activities.
- **Manual deployment** is the least sophisticated approach to release management. As the name suggests, someone on the team manually deploys the most recent code changes. This approach is error prone, unreliable, and considered an antipattern by most seasoned engineers.

During the first iteration of an MVP solution, manual deployment is common, despite the preceding assessment. When the solution is extremely fluid and customer feedback is unknown, there's a significant risk in resetting the entire solution (or even the core hypothesis). Here's the general rule for manual deployment: no customer proof, no deployment automation.

Investing early can lead to lost time. More importantly, it can create dependencies on the release pipeline that make the team more resistant to an early pivot. After the first few iterations or when customer feedback suggests potential success, a more advanced model of deployment should be quickly adopted.

At any stage of hypothesis validation, Azure DevOps and [Azure Pipelines](#) provide continuous delivery and continuous deployment capabilities. Learn more about [continuous delivery](#), or check out the [hands-on lab](#). Solution architecture can also accelerate creation of your [CI/CD pipelines through Azure DevOps](#).

Integrated measurements

When you [measure for customer impact](#), it's important to understand how customers react to changes in the solution. This data, known as *telemetry*, provides insights into the actions a user (or cohort of users) took when working with the solution. From this data, it's easy to get a quantitative validation of the hypothesis. Those metrics can then be used to adjust the solution and generate more fine-grained hypotheses. Those subtler changes help mature the initial solution in later iterations, ultimately driving to repeat adoption at scale.

In Azure, [Azure Monitor](#) provides the tools and interface to collect and review data from customer experiences. You can apply those observations and insights to refine the

backlog by using [Azure Boards](#).

Next steps

After you've gained an understanding of the CI/CD pipeline tools and processes needed to empower adoption, it's time to examine a more advanced innovation discipline: [interact with devices](#). This discipline can help reduce the barriers between physical and digital experiences, making your solution even easier to adopt.

[Interact with devices](#)

Ambient user experiences: Interact with devices

Article • 12/01/2022

In [Build with customer empathy](#), we discussed the three tests of true innovation: solve a customer need, keep the customer coming back, and scale across a base of customer cohorts. Each test of your hypothesis requires effort and iterations on the approach to adoption. This article offers insights on advanced approaches to reduce that effort through *ambient user experiences*. By interacting with devices, instead of an application, the customer might be more likely to use your solution first.

Ambient user experiences

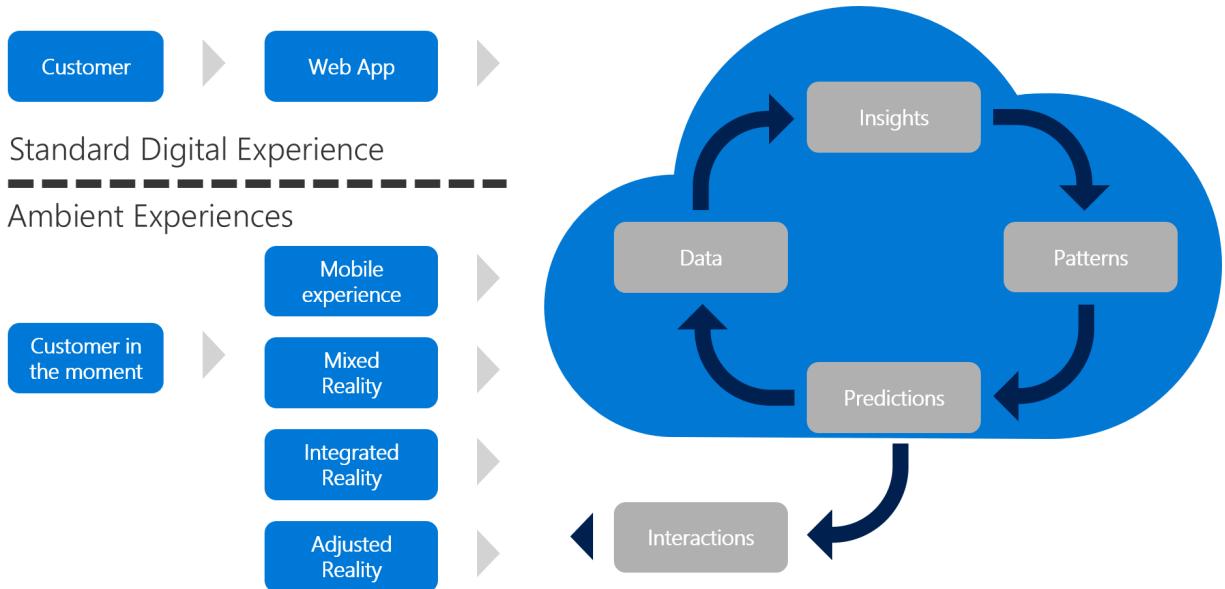
An ambient user experience is a digital experience that relates to the immediate surroundings. Ambient user experiences occur when technology systems seamlessly interact with user, based on their needs and the context of their requests. A solution that features ambient user experiences tries to meet the customer in the moment that they need it. When possible, the solution solves the customer's problem without leaving the flow of activity that triggered it.

Life in the digital economy is full of distractions. We're all bombarded with social, email, web, visual, and verbal messaging, each of which is a risk of distraction. This risk increases with every second that passes between the customer's point of need and the moment they find a solution. Countless customers are lost in that brief time gap. To foster an increase in repeat adoption, it's important to reduce the number of distractions by reducing the time to solution.

Interact with devices

A standard web experience is the most common application development technique you can use to meet a customer's needs. This approach assumes that your customer is in front of their computer. If your customer consistently meets their point of need while in front of their laptop, build a web application, which provides an ambient user experience for that scenario. However, this scenario is becoming less and less likely.

Interacting through devices & ambient experiences



These days, an ambient user experience typically requires more than a web application. Through **measurement** and **learning with the customer**, the behavior that triggers the customer's need can be observed, tracked, and used to build a more ambient and digital experience. The following list summarizes a few approaches to integration of ambient solutions into your hypotheses, with more details about each in the following paragraphs.

Types of interactive devices for ambient user experiences:

- **Mobile experience:** Mobile apps are ubiquitous in customer environments. In some situations, a mobile app might provide a sufficient level of interactivity to make a solution ambient.
- **Mixed reality:** Sometimes a customer's typical surroundings must be altered to make an interaction ambient. This factor creates something of a false reality in which the customer interacts with the solution and has a need met. In this case, the solution is ambient within the false reality.
- **Integrated reality:** Moving closer to true ambience, integrated reality solutions focus on the use of a device that exists within the customer's reality to integrate the solution into their natural behaviors. A virtual assistant is a great example of integrating reality into the surrounding environment. Another example is Internet of Things (IoT) technologies, which integrate devices that already exist in the customer's surroundings.
- **Adjusted reality:** When any of these ambient solutions use predictive analysis in the cloud to define and provide an interaction with the customer through the natural surroundings, the solution has adjusted reality.

Understanding the customer need and measuring customer impact help you determine whether a device interaction or ambient user experience are necessary to validate your

hypothesis. With those data points, the following sections will help you find the best digital experience solution.

Mobile experience

In the first stage of ambient user experience, the user moves away from the computer. Today's consumers and business professionals move fluidly between mobile and PC devices. Each of the platforms or devices your customer uses creates a new potential experience. Adding a mobile experience that extends the primary solution is the fastest way to improve integration into the customer's immediate surroundings. While a mobile device is far from ambient, it might edge closer to the customer's point of need.

When customers are mobile and change locations frequently, that can represent the most relevant form of ambient and digital experience for a particular solution. Over the past decade, innovation has frequently been triggered by the integration of existing solutions with a mobile experience.

Azure App Service is a great example of this approach. During early iterations, you can use the [web app feature of Azure App Service](#) to test your hypotheses. As your hypotheses become more complex, you can use a [mobile app](#) to extend the web app to run in various mobile platforms.

Mixed reality

Mixed reality solutions represent the next step for ambient user experiences. This approach augments or replicates the customer's surroundings; it creates an extension of reality for the customer to operate within.

Important

If a virtual reality (VR) device is required and it's not already part of a customer's immediate surroundings or natural behaviors, augmented or virtual reality is more of an alternative digital experience and less of an ambient experience.

Mixed reality experiences are increasingly common among remote workforces. Their use is growing even faster in industries that require collaboration or specialty skills that aren't readily available in the local market. Situations that require centralized implementation support of a complex product for a remote labor force are fertile ground for augmented reality. In these scenarios, the central support team and remote

employees might use augmented reality to work on, troubleshoot, and install the product.

For example, consider the case of spatial anchors. Spatial anchors allow you to create mixed reality experiences with objects that persist in their respective locations across devices over time. Through spatial anchors, you can capture, record, and persist a specific behavior, providing an ambient experience the next time a user operates within that augmented environment. [Azure Spatial Anchors](#) is a service that moves this logic to the cloud, allowing digital experiences to be shared across interactive devices and even across solutions.

Integrated reality

Beyond mobile reality, or even mixed reality, lies integrated reality. Integrated reality aims to remove the digital experience entirely. All around us are devices with compute and connectivity capabilities. These devices can be used to collect data from the immediate surroundings without the customer having to ever touch a phone, laptop, or VR device.

This digital experience is best when some form of device is consistently within the same surroundings in which the customer need occurs. Common scenarios include factory floors, elevators, and even your car. These types of large devices already contain compute power. You can also use data from the device itself to detect customer behaviors and send those behaviors to the cloud. This automatic capture of customer behavior data dramatically reduces the need for a customer to input data. Additionally, the web, mobile, or VR experience can function as a feedback loop to share what's been learned from the integrated reality solution.

Examples of integrated reality in Azure:

- [Azure Internet of Things \(IoT\) solutions](#): A collection of services in Azure that each aid in managing devices and the flow of data from those devices into the cloud and back out to end users.
- [Azure Sphere](#): A combination of hardware and software that provides an intrinsically secure way to enable an existing device to securely transmit data between the device and Azure IoT solutions.
- [Azure Kinect DK](#), AI sensors with advanced computer vision and speech models. These sensors can collect visual and audio data from the immediate surroundings and feed those inputs into your solution.

You can use all three of these digital experience tools to collect data from the natural surroundings and at the point of customer need. From there, your solution can respond

to those data inputs to solve the need, sometimes before the customer is even aware that a trigger for that need has occurred.

Adjusted reality

The highest form of ambient user experience is adjusted reality, often called *ambient intelligence*. Ambient intelligence refers to any electronic or digital environment that responds to the presence of people and adjusts automatically. Adjusted reality is an approach to using information from your solution to change the customer's reality without requiring them to interact directly with an application. In this approach, the application you initially built to prove your hypothesis might no longer be relevant at all. Instead, devices in the environment help modulate the inputs and outputs to meet customer needs.

Virtual assistants and smart speakers offer great examples of adjusted reality. Alone, a smart speaker is an example of simple integrated reality. But add a smart light and motion sensor to a smart speaker solution and it's easy to create a basic solution that turns on the lights when you enter a room.

Factory floors around the world provide additional examples of adjusted reality. During early stages of integrated reality, sensors on devices detected conditions like overheating, and then alerted a human being through an application. In adjusted reality, the customer might still be involved, but the feedback loop is tighter. On an adjusted reality factory floor, one device might detect overheating in a vital machine somewhere along the assembly line. Somewhere else on the floor, a second device then slows production slightly to allow the machine to cool and then resume full pace when the condition is resolved. In this situation, the customer is a second-hand participant. The customer uses your application to set the rules and understand how those rules have affected production, but they're not necessary to the feedback loop.

The Azure services described in [Azure Internet of Things \(IoT\) solutions](#), [Azure Sphere](#), and [Azure Kinect DK](#) can all be components of an adjusted reality solution. Your original application and business logic would then serve as the intermediary between the environmental input and the change that should be made in the physical environment.

A digital twin is another example of adjusted reality. This term refers to a digital representation of a physical device, presented through computer, mobile, or mixed-reality formats. Unlike less sophisticated 3D models, a digital twin reflects data collected from an actual device in the physical environment. This solution allows the user to interact with the digital representation in ways that could never be done in the real world. In this approach, physical devices adjust a mixed reality environment. However,

the solution still gathers data from an integrated reality solution and uses that data to shape the reality of the customer's current surroundings.

In Azure, digital twins are created and accessed through a service called [Azure Digital Twins](#).

Next steps

Now that you have a deeper understanding of device interactions and the ambient user experience or ambient intelligence tools that are right for your solution, you're ready to explore the final discipline of innovation, [predict and influence](#).

[Predict and influence](#)

Predictive modeling and influencing customer behavior

Article • 08/28/2024

There are two classes of applications in the digital economy: *historical* and *predictive*. Many customer needs can be met solely by using historical data, including nearly real-time data. Most solutions focus primarily on aggregating data in the moment. They then process and share that data back to the customer in the form of a digital or ambient experience.

In contrast to historical modeling is predictive modeling. But, what is predictive modeling? Predictive modeling uses statistics and known results to process and create models that can be used to predict future outcomes, within reason. As predictive modeling becomes more cost-effective and readily available, customers demand forward-thinking experiences that lead to better decisions and actions. However, that demand doesn't always suggest a predictive solution. In most cases, a historical view can provide enough data to empower the customer to make a decision on their own.

Unfortunately, customers often take a myopic view that leads to decisions based on their immediate surroundings and sphere of influence. As options and decisions grow in number and impact, that myopic view may not serve the customer's needs. At the same time, as a hypothesis is proven at scale, the company providing the solution can see across thousands or millions of customer decisions. This big-picture approach makes it possible to see broad patterns and the impacts of those patterns. Predictive modeling capability is a wise investment when an understanding of those patterns is necessary to make decisions that best serve the customer.

Examples of predictive modeling and how it influences customer behavior

Various applications and ambient experiences use data to make predictions:

- **E-commerce:** Based on what other similar consumers have purchased, an e-commerce website suggests products that may be worth adding to your cart.
- **Adjusted reality:** IoT offers more advanced instances of predictive functionality. For example, suppose a device on an assembly line detects a rise in a machine's temperature. A cloud-based predictive model determines how to respond. Based on that prediction, another device slows down the assembly line until the machine can cool.

- **Consumer products:** Cell phones, smart homes, even your car, all use predictive capabilities, which they analyze to suggest user behavior based on factors like location or time of day. When a prediction and the initial hypothesis are aligned, the prediction leads to action. At a very mature stage, this alignment can make products like a self-driving car a reality.

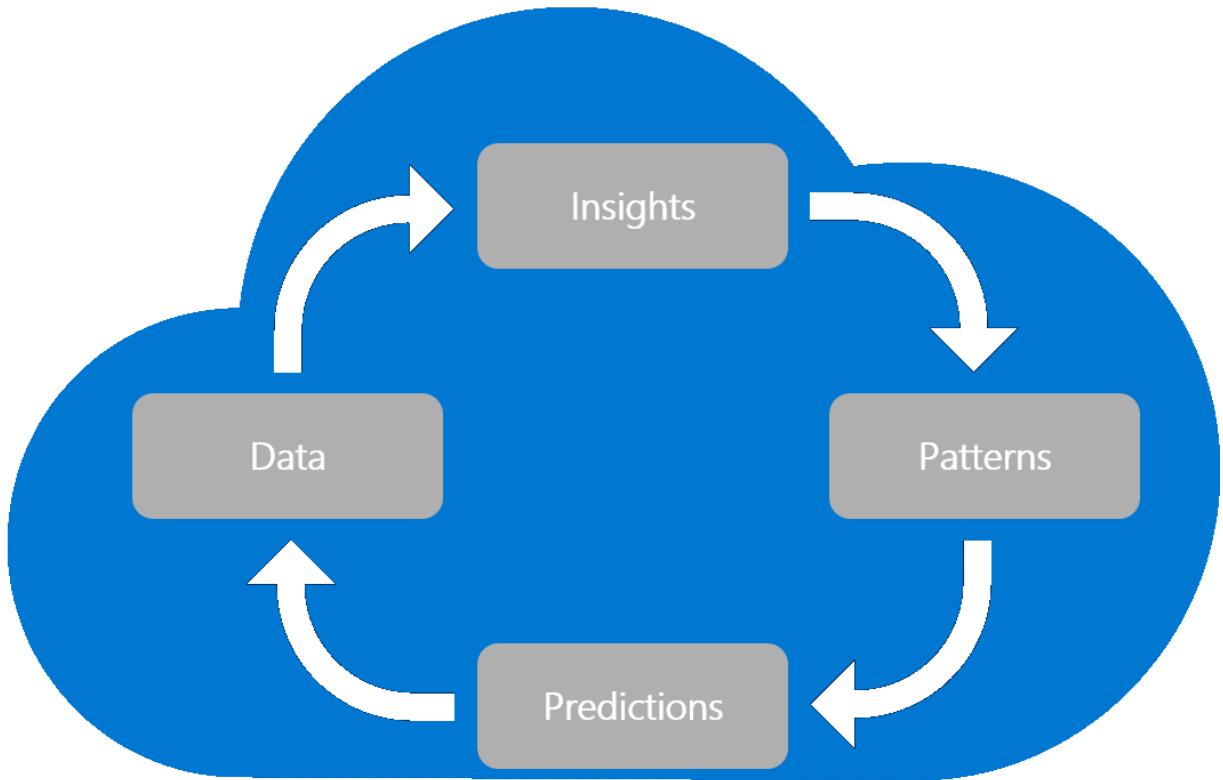
Develop predictive capabilities

Solutions that consistently provide accurate predictive capabilities commonly include five core characteristics. The five core predictive modeling characteristics are:

- Data
- Insights
- Patterns
- Predictions
- Interactions

Each aspect is required to develop predictive capabilities. Like all great innovations, the development of predictive capabilities requires a [commitment to iteration](#). In each iteration, one or more of the following characteristics is matured to validate increasingly complex customer hypotheses.

Predict & Influence



Interactions

⊗ Caution

If the customer hypothesis developed in [Build with customer empathy](#) includes predictive capabilities, the principles described there might well apply. However, predictive capabilities require significant investment of time and energy. When predictive capabilities are [technical spikes](#), as opposed to a source of real customer value, we suggest that you delay predictions until the customer hypotheses have been validated at scale.

Data

Data is the most elemental of the characteristics mentioned earlier. Each of the disciplines for developing digital inventions generates data. That data, of course, contributes to the development of predictions. For more information on ways to get data into a predictive solution, see:

- Democratize data with digital invention
- Interact with devices

Various data sources can be used to deliver predictive capabilities:

Insights

Subject matter experts use data about customer needs and behaviors to develop basic business insights from a study of raw data. Those insights can pinpoint occurrences of the desired customer behaviors (or, alternatively, undesirable results). During iterations on the predictions, these insights can aid in identifying potential correlations that could ultimately generate positive outcomes. For guidance on enabling subject matter experts to develop insights, see [Democratize data with digital invention](#).

Patterns

People have always tried to detect patterns in large volumes of data. Computers were designed for that purpose. Machine learning accelerates that quest by detecting precisely such patterns, a skill that comprises the machine learning model. Those patterns are then applied through machine learning algorithms to predict outcomes when a new set of data is entered into the algorithms.

Using insights as a starting point, machine learning develops and applies predictive models to capitalize on the patterns in data. Through multiple iterations of training, testing, and adoption, those models and algorithms can accurately predict future outcomes.

[Azure Machine Learning](#) is the cloud-native service in Azure for building and training models based on your data. This tool also includes a [workflow for accelerating the development of machine learning algorithms](#). This workflow can be used to develop algorithms through a visual interface or Python.

Predictions

After a pattern is built and trained, you can apply it through APIs, which can make predictions during the delivery of a digital experience. Most of these APIs are built from a well-trained model based on a pattern in your data. As more customers deploy everyday workloads to the cloud, the prediction APIs used by cloud providers lead to ever-faster adoption.

Azure Machine Learning lets you deploy custom-built algorithms, which you can create and train based solely on your own data. For information about deploying predictions with Azure Machine Learning, see [Deploy machine learning models to Azure](#).

Interactions

After a prediction is made available through an API, you can use it to influence customer behavior. That influence takes the form of interactions. An interaction with a machine learning algorithm happens within your other digital or ambient experiences. As data is collected through the application or experience, it's run through the machine learning algorithms. When the algorithm predicts an outcome, that prediction can be shared back with the customer through the existing experience.

Learn more about how to create an ambient experience through an [adjusted reality solution](#).

Next steps

Review a prescriptive framework that includes the tools, programs, and content (best practices, configuration templates, and architecture guidance) to simplify adoption for the following innovation scenarios.

- [Kubernetes in the Cloud Adoption Framework](#)
-

Feedback

Was this page helpful?

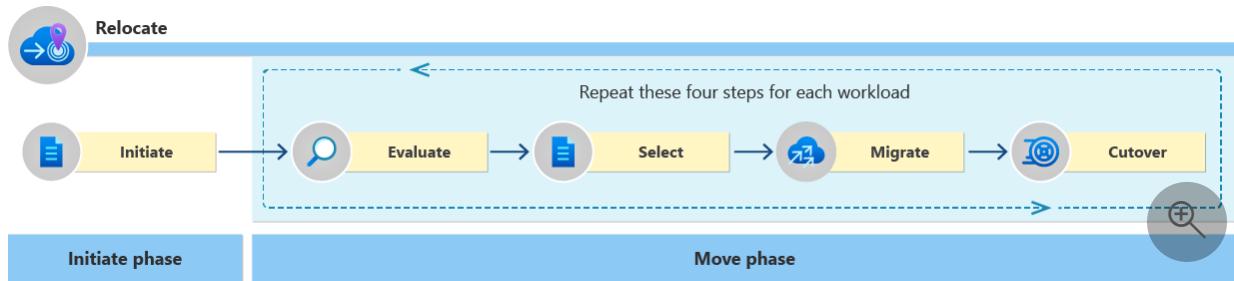
 Yes

 No

Relocate cloud workloads

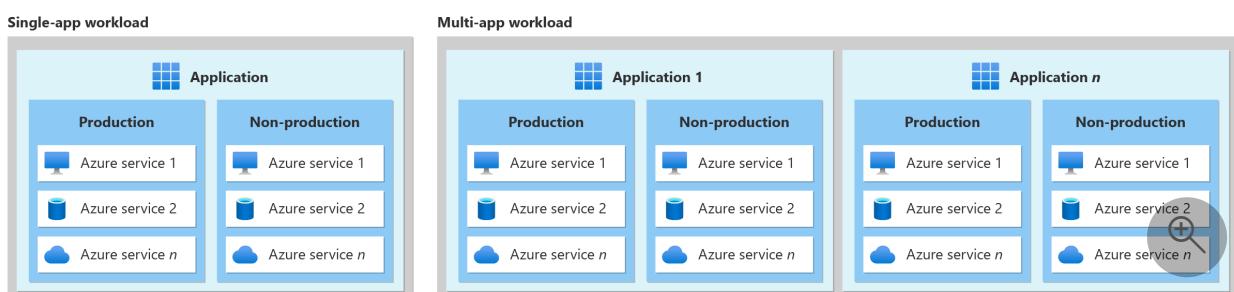
Article • 12/20/2023

The relocate guidance shows you how to set up a relocation project and relocate one or more workloads.



What is relocation?

Relocation is the process of moving a workload or workload component in Azure to a different Azure region. A workload is a collection of applications, environments, services, and data that support a defined process. A workload can have one or many applications. Relocating a single or multi-app workload to a different region is a type of migration and has similarities with the process defined in the [migrate guidance](#). But relocation also has its own solutions and considerations to implement, and the relocate guidance here outlines these distinctive features to help you navigate the relocation process.



Why relocate?

Relocation adds flexibility that can help you optimize cost and performance throughout the lifecycle of your workloads. When workloads are first deployed or migrated to Azure, you make a decision about their region. However, as time passes, you should review that decision to see if a different region might be a better fit. For example, another region could have services or capabilities unavailable in the current region, or you might want to move your workload closer to a new customer base. Data residency laws might change, or budgets could shift. Rather than work around these changes and encounter

cost or performance issues, a relocation might be the best way to proceed for your workload.

[+] [Expand table](#)

Relocation drivers	Examples
Business developments	Respond to business changes and expand your global footprint.
Compliance	Meet data sovereignty and residency requirements.
Proximity	Provide lower latency to end users.

How to relocate

Relocation has two phases. The first phase is to initiate the relocation project. The second phase is to move the workload by planning and executing the relocation. Here's an overview of each phase:

- *Initiate phase:* The initiate phase has a single-step also called Initiate. The goal of the Initiate phase is to set up the relocation project, get stakeholder approval, and identify workloads for relocation.
- *Move phase:* The Move phase is a four-step process to plan and move a workload to a different region. The steps in the Move Phase are (1) evaluate, (2) select, (3) migrate, and (4) cutover. After you cut over the final workload, you need to officially close to the relocation project.

[+] [Expand table](#)

Relocate steps	Main goal
1. Initiate	Establish relocation project.
2. Evaluate	Conduct workload discovery.
3. Select	Pick the right relocation method.
4. Migrate	Relocate the workload.
5. Cutover	Direct traffic to new location.

These phases define the lifecycle of a relocation project. It's important to note that not every resources supports relocation. Some require redeployment. For more information, see [Move operation support for resources](#).

Assumptions

You should read relocation best practices and use them as guidance for each workload relocation. The following articles detail how you should approach each step. The content highlights critical areas to focus on but also refers you to other documentation for deeper technical insight. It assumes a basic understanding of Azure regions and service availability. For more information, see:

- [Azure regions selection guide](#)
- [Special conditions to consider when moving a resource](#)
- [Azure Products by Region](#) ↗
- [Azure regions and availability zones](#)
- [List of region pairs](#)
- [Azure Services](#)
- [How to move resources](#)

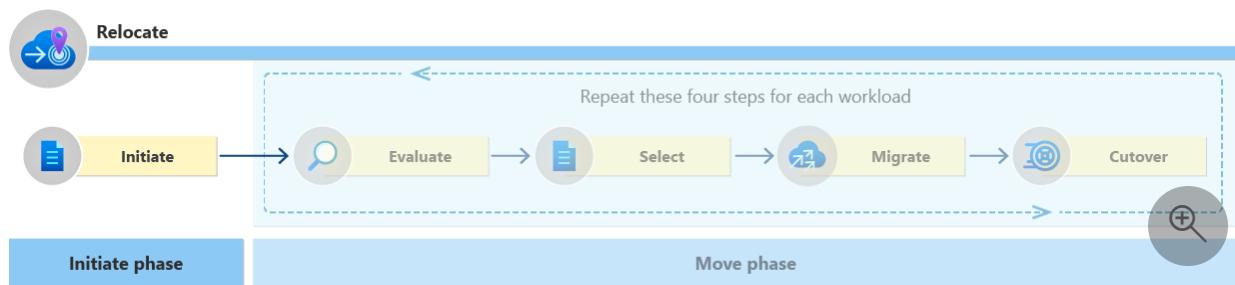
Whether you're relocating multiple workloads or just one, you can aid the success of relocation with some formal relocation planning. The Initiate phase provides required guidance.

Initiate

Initiate a cloud relocation project

Article • 12/20/2023

The Initiate phase is a single step to establish the relocation project. The guidance in the Initiate phase is more important for large relocation efforts with multiple workloads in scope. For small relocation with a single workload in scope, you can skip to [create a relocation plan](#). A single relocation project can include multiple source and target regions. There's no need to plan them separately. The goal of the Initiate phase is to provide stakeholders with the information they want and get the relocation the resources it needs. Here's a framework for initiating a relocation.



Identify objectives

The need to meet specific objectives drives relocation projects. You should capture these objectives so you can scope and prioritize your relocation efforts. The following table provides examples of relocation objectives.

[Expand table](#)

Objective	Motivation	Priority
1. Comply with data-residency laws.	Meet all legal and ethical standards.	High
2. Improve application performance for users in a different region.	Meet revenue targets.	Medium
3. Add new services to internal productivity tools.	Increase employee productivity.	Low

Determine relocation scope

You should use the objectives to scope your relocation plan. Identify all the workloads in scope of these objectives. The workload should be the smallest unit of relocation during

scoping. Anything smaller, such as environment or service, makes planning difficult to manage. In execution, you relocate workloads by migrating or redeploying workload services and components. But for planning and scoping, focus on the workload as a whole. For more information, see [What is a workload?](#).

Prioritize workloads for relocation

The order you relocate workloads should reflect the priority of your objectives. Any technical dependencies, such as platforms and landing zones, should go first. Outside of technical dependencies, the order is up to you. For more information, see [Platform vs application landing zones](#).

Create a relocation plan

You should create a relocation plan that addresses service capacity and high-level project planning. At the service level, you want to verify the target region meets the capacity needs for each workload service. Once you prioritized the workloads, the rest of your relocation plan should center on the relocation project and getting stakeholder approval to proceed. Most stakeholders want to know about the schedule, team, and cost. Rather than spending time trying to estimate the schedule, resources, and cost, start the relocation process and refine the estimates as you learn more. Here are recommendations for creating a minimum viable plan to get started.

- *Build a soft schedule.* The schedule depends on the complexity of the workloads being relocated and the experience of the team involved. Instead of spending time figuring out how long it takes, start relocating the first workload. The experience allows you to better estimate a completion date, and it saves you time.
- *Use a multi-disciplinary team.* The relocation team needs technical, business, and legal/compliance expertise. The business stakeholders own the workload and should guide the technical team to meet the objectives. You need the legal or compliance teams if the workload is being relocated due to regulatory requirements.
- *Know cost factors.* Relocation has a cost. Moving data and using services isn't free. To minimize cost, you should move data once and avoid duplicating services or environments for extended periods of time. A hidden cost you might not consider comes from suppliers, partners, or third-party technical support. You might need their help to relocate a workload, and they typically charge a fee.

The pricing for certain Azure services can differ based on the region. Therefore, it's crucial to review the costs associated with each region before initiating a relocation project. If you're currently utilizing Azure Reservations in your source region, consider [transferring them to your destination region](#). Also, don't forget to account for the [Azure network bandwidth costs ↗](#) associated with data transfers between regions during the relocation process.

Next step

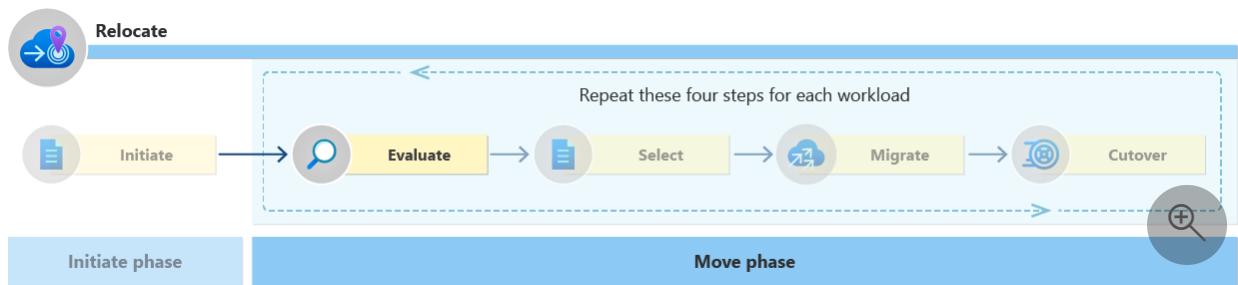
The next step is the Move phase of relocation. The Move phase has four steps: evaluate, select, migrate, and cutover. You might have one or more workloads to relocate. Regardless of number, you should work through these four steps for any workload you want to relocate.

[Evaluate](#)

Evaluate a cloud workload for relocation

Article • 12/20/2023

Evaluate is the first step in the Move phase of relocation. The goal of Evaluate is to understand the workload you want to relocate so you can move it successfully. Every workload you relocate must go through the four steps of the Move phase, starting with the Evaluate step.



Pick workload(s)

You should have a prioritized list of workloads, and the list should identify the order you want to relocate your workloads. Each time you visit the Evaluate step, pick the workload(s) at the top of the list. For smaller teams, you should relocate one workload at a time. It's a chance to learn and improve with each workload relocation. Larger teams should consider relocating multiple workloads. Bulk relocations can help achieve economies of scale.

Conduct discovery

Workload discovery is the foundation of relocation. The goal of discovery is to understand the workload enough to ensure a smooth relocation. Discovery must comprehensively uncover the organizational and technical dimensions of the workload.

Conduct organizational discovery

Workload organizational discovery involves finding out who is in charge of a workload, understanding the risks of moving it, and planning how to communicate about the move. This step helps identify who needs to be involved, manage risks, and ensure everyone is informed properly. This careful planning helps make the transition smoother and less disruptive for the business and its customers.

- *Workload ownership:* Determine who is responsible for the workload.

- *Stakeholder identification:* Identify all parties with an interest in the workload.
- *Risk assessment:* Assess the potential business risks associated with moving the workload.
- *Change management:* Gain a clear understanding of the processes for managing changes to the workload.
- *Outage windows:* Find out the acceptable times when the system can be offline for relocation.
- *Impact analysis:* Recognize which internal users or external customers the relocation might affect.
- *Communication needs:* Understand the current and needed communication plan to communicate any downtime or changes.
- *Policy compliance:* Ensure the relocation adheres to organizational policies and industry regulations.

Conduct technical discovery

Technical workload discovery involves comprehensively understanding the technical aspects of a workload, including its dependencies, resources, network configuration, and any other technical requirements or constraints. Technical discovery helps you anticipate and mitigate risks associated with the relocation. Here are strategies for conducting technical discovery.

Evaluate dependencies

Dependencies are resources or services that the workload needs to run. Identifying all dependencies is essential for a successful workload relocation. Dependencies encompass various resources and services essential for the workload's operation. The following list provides a few examples of dependencies:

- *Azure services:* Any Azure services that the workload relies on. Global resources don't deploy to a specific region, so you don't need to move them in a relocation. However, you might still reconfigure them to work in a different region. For example, you might need to update IP addresses in Azure Front Door profile to point to the new IP address of a relocated workload.
- *Non-Microsoft applications:* Applications from other vendors that are integrated with or necessary for the workload. Understand the features and any limitations of

the products involved in the workload.

- *Licenses*: Ensure that all required software licenses are accounted for and remain valid in the new location.
- *Networking*: Understanding the network setup, including firewalls, to ensure seamless connectivity post-relocation.
- *Testing*: Determine the testing procedures needed to ensure the workload functions correctly in the new environment.
- *Tagging*: Properly tag and identify resources for effective management and tracking.
- *Automation*: Your organization might use scripts and infrastructure as code. You must update any references to Azure regions, service names, or service URLs within the scripts or code. The references need to correspond to the new Azure region you're moving to.

You should avoid hardcoding any values in your code that are subject to change during the workload lifecycle. Instead, retrieve these values dynamically or use configurable parameters within the code. This approach makes changes less burdensome and ensures a smoother relocation process.

- *DNS*: Azure assigns public IP addresses to endpoints depending on the region. When you move an endpoint to a different Azure region, it gets a different IP address. Make sure to update your DNS records with these new IP addresses. Also, you need to provide the new IP address to any system that has the former IP address on its 'allowed' list.

You might need to deploy new resources in the new Azure region before you turned off the old ones. If so, you could run into issues where two resources can't have the same DNS name at once. Think about using unique names for each service to avoid this problem. In some cases, you might be able to use CNAME records to provide a layer of abstraction. It makes resource name changes easier to manage.

- *Load balancers*: Update load balancers to point to any new backend IP addresses or hosts. For DNS-based load balancers, the change might take some time to propagate based on DNS caches and time-to-live (TTL) record expiry. For more information, see [Azure load balancing services](#). Consider temporarily decreasing the Time to Live (TTL) settings for your DNS records. It helps the DNS records switch to new IP address faster. Also, consider setting your load balancer to check the health of your backend systems more frequently for a short period. Remember

to change these settings back to normal after the migration to avoid extra costs and reduced performance later on.

- *Azure Backup registration.* When you move virtual machines to a new Azure region, make sure to unregister them from the Azure Backup service in the current region and register them with the Azure Backup service in the new region. You can't access the existing backup recovery points because they can't be transferred to the new backup vault. You need to start creating new recovery points in the new region.

Evaluate endpoints

Endpoint discovery refers to the process of identifying all the endpoints or IP addresses associated with a workload. Discovering all workload endpoints ensures you account for all network connections and access points and prepare to properly configure them in the new environment. Here are recommendations for dealing with public IP addresses and private endpoints:

- *Public IP addresses:* Public IP addresses are region specific. You can't move them between regions. You need to export the configuration of a public IP and deploy it to the new target region. For more information, see [Move Azure Public IP configuration to another Azure region](#).
- *Private endpoints:* When you redeploy a private endpoint, it's likely gets a new IP address from the subnet you link it to. If you connect to your resources through a private endpoint, these endpoints link to private DNS zones that resolve the resource's network address within your virtual network. In a relocation, you need to update the DNS records within the private DNS zones to maintain connectivity.

Use automated tools

Where possible, use automated tools to collect information about applications and Azure services that make up your workload. You can use these tools to perform low-level discovery and architecture design discovery for the relocation of a specific workload. You should use the following Azure tools and services.

Try Azure Resource Mover. You should try Azure Resource Mover first. It's currently the easiest discovery tool to use, and you can also relocate services and data with the service. However, Azure Resource Mover only supports a limited number of services, so make sure your services are supported before continuing. For more information, see [Supported resources for Azure Resource Mover](#).

Use visualization tools. If Azure Resource Mover doesn't meet all your needs, you can use visualization tools to aid your discovery. Azure has several visualization tools that you can use to map dependencies. Pick the tool that best supports your needs.

- *Resource group visualizer*: You can visualize the connections between the resources in a resource group. In the Azure portal, navigate to the resource group and select *Resource visualizer* from the left navigation.
- *Azure Monitor topology*: You can view network dependencies with the topology feature in Azure Monitor. For more information, see [Network insights topology](#).
- *Application Insights*: Application Insights has an application mapping feature where you can view the logical structure of a distributed application. For more information, see [Application map in Azure Application Insights](#).
- *Azure Resource Explorer*: Azure Resource Explorer lists every resource in your Microsoft Entra tenant. It gives you visibility but doesn't indicate dependencies. You must map workload components and dependencies manually. For more information, see [Azure Resource Explorer](#).
- *Azure Resource Graph*: Azure Resource Graph allows you to run queries against the resources in a Microsoft Entra tenant. Resource Graph is accessible in the portal and from the command line. You must map workload components and dependencies manually. For more information, see [Azure Resource Graph documentation](#).
- *Inventory dashboard*: In the Azure portal, you can use the built-in inventory template to create a dashboard to track your existing resources. It's a quick way of determining the resources you have and the number of instances.

Manually create documentation

If automated discovery approaches aren't enough, you can conduct a manual assessment of the workloads. Most manual assessments rely on interviews with technical experts and technical documentation to get the information needed. Identify product or application owners and interview them. These interviews are optional, but necessary when the team needs to cover gaps in the information the tools provide. And the app owner can pull the tags and manually identify dependencies.

Find region supportability

Not every region in Azure offers the same services, so you must make sure the services your workload needs to run are available in the target region. It might seem late in the process to make this determination, but you need the discovery details to ensure supportability. To determine region supportability for your workload, see the [products and services available in each Azure region](#).

Know if the target region is a paired region or not and if it supports availability zones. Region pairing and availability zones don't affect the relocation effort, but they do affect your business continuity and disaster recovery (BCDR) strategy in the target region. For more information, see [Azure geographies](#) and [Availability zones](#).

Categorize workload services

Relocation happens at the service and component level. Most workloads use multiple services. There are two primary types of services, stateful and stateless. You need to categorize each service as stateful or stateless. This knowledge helps you determine dependencies, understand service integrations, and narrows your relocation automation options.

- *Stateless services*: Stateless services have configuration information only. These services don't need continuous replication of data to move. Examples include virtual networks, network adapters, load balancers, and network security groups.
- *Stateful services*: Stateful services have configuration information and data that need to move. Examples include virtual machines and SQL databases.

Next steps

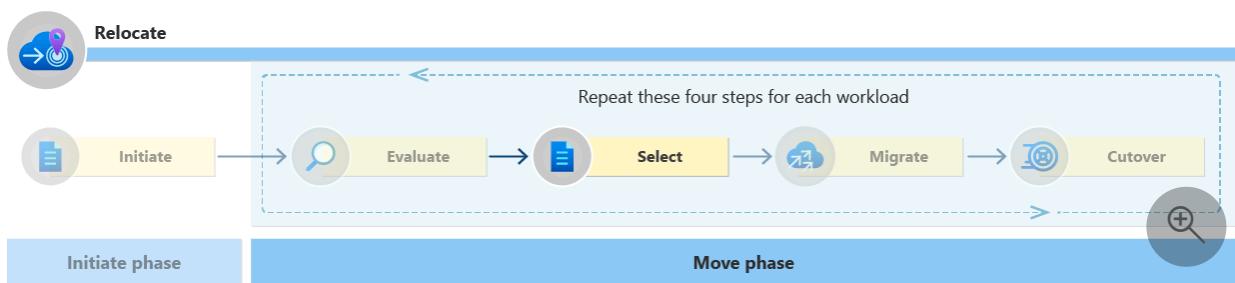
Evaluating your workload provides enough information to select a relocation method and the tools to execute the method you choose. The Select step walks you through the decisions to pick a relocation method and the correct tools for the relocation method.

Select

Select a relocation strategy for cloud workloads

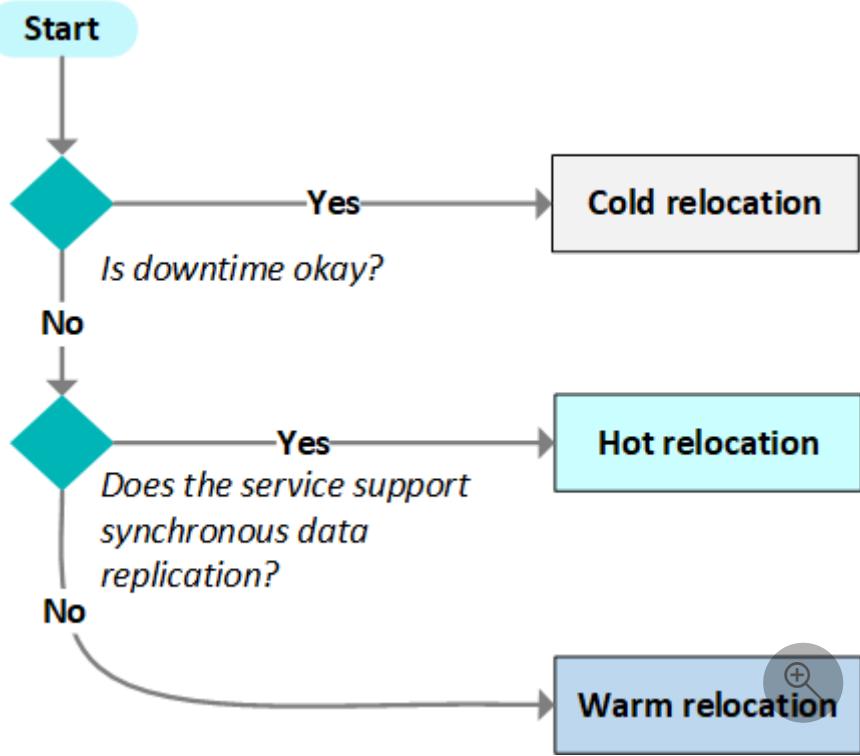
Article • 06/05/2024

Before you start migrating the workload to another region, you need to plan your relocation strategy. The strategy includes the relocation method, service-relocation automation, and data-relocation automation. This article lays out the options for each strategy component and guides you toward a decision. Ultimately, the selections you make depend on the services and the criticality of the workload.



Select a relocation method

There are three main methods for relocating workloads. The relocation method you choose depends on the services in the workload and how critical the workload is to essential business functions. You could consider different relocation methods for production and nonproduction environments. Cold relocation is for nonessential workloads. Hot and warm relocation is for mission-critical. The method you choose relocation affects service and data relocation tools you use to relocate the workload. Use the following relocation decision tree to get a general idea of the right relocation method and validate your decision by reading the overview of the three relocation methods.



Cold relocation

Cold relocation is for workloads that can withstand downtime. It's the most cost-effective approach to relocation because you don't duplicate any environments during relocation. Here's an overview of the cold relocation process.

1. Back up workload data to the new target region.
2. Take the source region offline and shut down services.
3. Deploy the cloud services to the new target region.
4. Restore workload data.

Cold relocation can take a few minutes or a few days depending on the number of services and volume of data.

Hot relocation

The hot relocation method is for workloads that need minimal (seconds, minutes) to zero downtime. For critical workloads, you should see if the service supports hot relocation before trying a warm approach. Hot relocation helps minimize the data delta after cutover. Hot relocation is only possible if the service supports synchronous data replication. Some services don't have this feature, and you need to use a warm relocation approach instead. Here's the hot relocation process.

1. Perform service replication in the new target region.
2. Keep the workload running in the source region.

3. Start synchronous data replication.
4. After the data synchronizes, activate and validate the endpoints.
5. Stop the data synchronization.
6. Shut down the service in source region.

Warm relocation

Warm relocation is for critical workloads that don't support hot relocation. Warm relocation uses asynchronous data replication and environment replication. Here's the warm relocation process.

1. Perform service replication in the new target region.
2. Keep the workload running in the source region.
3. Create a backup of the source data. It's a best practice to create the backup during off-peak hours. You should also enable data-in replication to synchronize the data and minimize the data delta.
4. Restore the data in the new target region.
5. Switch and validate endpoints.
6. Shut down the workload in the source region.

Warm relocation can take a few minutes or an hour depending on the number of services and volume of data.

Select service-relocation automation

There are two primary service-relocation automation methods: infrastructure as code (IaC) and Azure Resource Mover. Each Azure service supports one or both automation approaches. Use the [Azure services relocation guidance](#) to see which automation method each Azure service supports and detailed steps for relocation. Here's an overview of the automation that the service relocation guidance uses:

- *Infrastructure as code (IaC)*: IaC can relocate every Azure service. Export the Azure Resource Manager (ARM) template (JSON) of an existing Azure service. Modify the template as needed and redeploy the template to a new region. You can convert ARM templates to Bicep templates by [pasting the JSON](#) into Visual Studio Code. When you use IaC to deploy a new instance of an Azure service, you can deploy multiple copies of the resource in parallel. With multiple copies, you can use one of the cutover techniques to redirect connections to the workloads in the new target region. Infrastructure as code (IaC) doesn't relocate data. Data relocation requires extra steps to move data to the newly deployed resource in the target region. Use the [data-relocation automation](#) guidance for more details.

- *Azure Resource Mover*: Azure Resource Mover allows you to move a limited number of [supported Azure resources](#) with its dependencies between regions, subscriptions, and resource groups.

Select data-relocation automation

If you used IaC to relocate stateful Azure services, you need to use a data-relocation automation method to relocate your data. For data relocation, you need to have the Azure service running in the target region before moving the data. Review the [relocation methods](#) to get a sense of the relocation sequence and where data relocation fits. Here's a list of automation tools you can use to relocate data:

- *Synchronous data replication*: Synchronous data replication replicates data in near real-time across regions. It's the preferred data relocation approach for hot relocation because it limits downtime and data delta migrations after cutover. This capability is built into some Azure services such as [Data Sync in Azure SQL Database](#). You need to check each service in your workload to see if it supports synchronous data replication.
- *Geo-replication*: Geo-replication can be a useful data relocation tool for the Azure services that support it. The way a geo-replication feature handles data and the underlying service instance varies across supported Azure services. Before using geo-replication for data relocation, you need to understand the geo-replication feature of the particular service you're relocating. For examples, see [Azure SQL](#) and [Cosmos DB](#).
- *Azure Site Recovery*: Azure Site Recovery can relocate services and data. It supports warm and cold relocation. For more information, see [Azure Site Recovery overview](#).
- *AzCopy*: AzCopy is a command-line utility that automates data movements in and out of Azure Storage. You need to download the tool and then use Microsoft Entra ID or shared access signature (SAS) tokens to authorize the move. For more information, see [AzCopy overview](#) and [Use AzCopy](#).
- *Pipelines and activities in Azure Data Factory or Synapse Analytics*: Azure Data Factory is a fully managed cloud-based data integration service that orchestrates and automates the movement and transformation of data. Azure Data Factory pipelines can move data lakes and warehouses. Synapse Analytics copy activity can also move data. For more information, see [Supported targets and sources](#) and [Copy data tool](#).

- *Azure Storage Explorer*: Azure Storage Explorer is a standalone app that allows you to relocate Azure Storage data. For more information, see [How to use Storage Explorer](#).
- *Azure Backup*: With Azure Backup, you can back up and restore data in another region. You should try Azure Backup first for nonessential cold and warm relocations. Azure Backup provides application-consistent, file-system consistent, and crash-consistent backups for virtual machines. It also supports managed disks, files shares, and blobs. You can't transfer existing backup restore points to the new target region. Consider keeping the vault in your source region until the backups are no longer required. For more information, see [Azure Backup overview](#).
- *Manual backup and restore*: Backup and restore here refers to a process, not a specific tool. Many services in Azure provide redundancy options that let you back up data to a separate region and restore it manually. You need to perform a manual backup and restore for specific services like Azure Key Vault. For more information, see [Move Key Vault to another region](#).

[] Expand table

Tool	Relocation method
Synchronous data replication	Hot, Warm
Geo-replication	Hot, Warm
Azure Site Recovery	Warm, Cold
AzCopy	Warm, Cold
Pipelines and activities in Azure Data Factory or Synapse Workspace	Warm, Cold
Azure Storage Explorer	Warm, Cold
Azure Backup	Cold
Manual backup and restore	Cold

Select cutover approach

Cutover is when you transition from the old workload to the new one. You direct traffic to the workload in the target region and no longer to the source region. The domain name system (DNS) is central to this redirection. As a reminder, DNS tells browsers and API clients where to get a response. It resolves domain names to IP addresses. Every domain needs a domain host to manage it. Azure DNS is the Azure domain host service.

There are different approaches to workload cutover, and the approach you take depends on the services in your workload. Here are a few examples.

- *Azure DNS*: For domains hosted in Azure DNS, you can perform a manual cutover by switching the CNAME. This approach is a business continuity failover process that works for cutover. For more information, see [Manual cutover using Azure DNS](#).
- *Traffic Manager*: It's also possible to use a routing service like Traffic Manager for cutover and route workload traffic to different endpoints. Traffic Manager is a DNS-based routing service. For more information, see [Configure DNS names with Traffic Manager](#).
- *App Service*: Application-layer services, such as Azure App Service, have features that allow you to update the domain name. For more information, see [Migrate an active DNS name to Azure App Service](#).
- *Gateway routing*: If the workload uses the [Gateway Routing pattern](#) with a service, such as Azure Front Door, Application Gateway, or Azure API Management, you can often make a region migration cutover. You use their backend targets and routing-rules features.

Next step

You selected a relocation method and the tools to relocate your workload. Move on to the Migrate step to execute the relocation using these tools.

[Migrate](#)

Feedback

Was this page helpful?

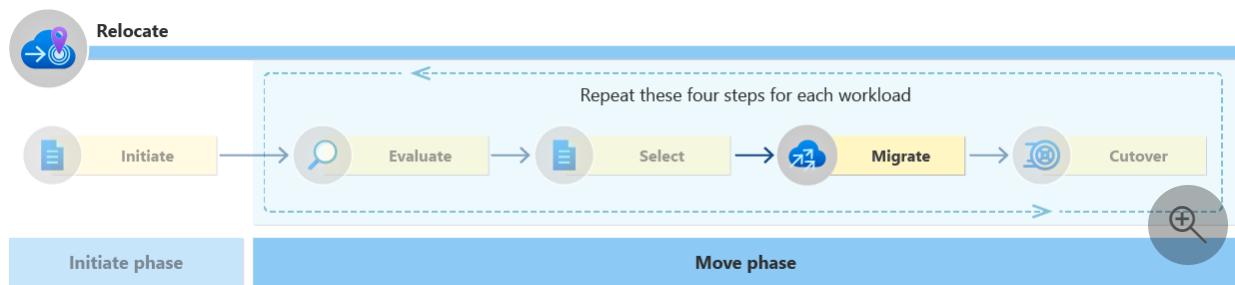
 Yes

 No

Migrate a cloud workload to another region

Article • 01/02/2024

The Migrate step of relocation is where you move the workload to the new region. Depending on your workload, you might have a few technical requirements to prepare, but the relocation strategy for the workload should be clear. You should be ready to execute the relocation.



Prepare

Before starting the workload relocation, you need to prepare the target region. As needed, follow these steps to prepare your workload environment before relocation. Doing so ensures you have core regional networking in place such as a regional hub and, if necessary, cross-premises connectivity.

Establish a landing zone. When planning your move, assess whether it expands the scope of your Azure landing zone. If expansion is necessary, consult the Azure landing zone guidance as a foundational step. This step ensures your approach aligns with established best practices. For more information, see [Add a new region to an existing Azure landing zone](#). Important considerations for setting up your new landing zone include:

- **Networking:** Evaluate the network structure, routing paths, and connectivity requirements for the landing zone in the new region.
- **Integration:** Determine if there's a need to integrate the new landing zone with the one in your source region.
- **Selective resource relocation:** Decide if all resources move to the new region. If some resources remain in the original location, plan for a sustainable cross-region network topology to manage these distributed resources effectively.

Create new subscriptions only if needed. Only create new subscriptions if you need to restructure the services and resources involved. Try to keep the workload in its existing

subscription if possible because creating a new subscription adds complexity. Subscriptions serve as boundaries for budgets, policies, and role-based access controls (RBACs). For any new subscription, you need to validate budgets, policies, and RBACs. If you don't move all the resources in a subscription, then you need to rescope the identity and security policies to match the smaller grouping of resources. To create a new subscription, you need to create, scope, and implement the required Azure policies and RBAC roles in the target subscription. The goal is to maintain the governance and security posture.

Configure a new domain name if needed. When there's a change in the custom domain of the workload, you need to configure a new domain name. Create the new hostname, assign it to your application or service, and then validate the name resolution. You might also plan to lower and configure the time-to-live (TTL) and set it in the cutover stage for auto expiry. For more information, see [Add your custom domain](#) and [Map DNS name to App Service plan](#).

Create new SSL/TLS certificates if needed. You need to create new SSL/TLS certificates (X.509) for any new domain name. These certificates enable public-private key encryption and secure network communication (HTTPS). Use Azure Key Vault to create or import X.509 certificates. For more information, see [Azure Key Vault certificates](#) and [Certificate creation methods](#)

Relocate Azure Key Vault. You should relocate Azure Key Vault before moving your workload. You should have one key vault per application environment, and your key vault shouldn't share secrets across regions to ensure confidentiality. You might need to create a new key vault in the new target region to align with this guidance.

Create a new Log Analytics workspace. You should have a separate Log Analytics workspace for each region. Create a new workspace in the target region. Since can't move a Log Analytics Workspace to another region, you need to create a new Log Analytics workspace in the target region. There are two options to preserve the data in the original workspace. You can keep the current workspace until you don't need the data, treating the data as read-only. You can also export the workspace data to a storage account in the new target Azure region.

Migrate services

You can begin migrating the services in your workload. For execution, follow available guidance for the relocation automation you selected. Azure Resource Mover and Azure Site Recovery have step-by-step tutorials to follow for service relocation. For more information, see:

- [Azure Resource Mover tutorials](#)
- [Azure Site Recovery tutorials](#)

You can create infrastructure-as-code templates or scripts for resources you can't move. You can also execute deployments manually in the Azure portal. The specific steps you follow depend on the Azure services you use and their configuration. For more information, see [Infrastructure as code overview](#).

Migrate data

This stage is only relevant when the workload requires data migration. Perform data migration with the chosen automation. Before the cutover to the workload in the target region, you must ensure that the related data is in sync with the source region. Data consistency is an important aspect that requires care. Here's guidance for migrating workload data.

1. *Migrate source region data.* The approach to migrating source-region data should follow the relocation method for the workload service. The hot, cold, and warm methods have different processes for managing the data in the source region.
2. *Synchronize data.* The synchronization technique depends on the architecture of the workload and the demand of the application. For example, in an on-demand sync, changes are pulled when data is accessed for the first time. Pulling and merging of changes occurs only in cases where there's a difference between last and current application access.
3. *Resolve synchronization conflicts.* Make sure the data in the source and target location are in sync and resolve any data conflicts. You only want users trying to access data that is available. For example, Azure Cosmos DB can serialize concurrent writes to ensure data consistency.

Update connection strings

The connection string configuration depends on the service the application connects to. You can search for "connection string" on our documentation page to find the service-specific guidance and use that guidance to update the connection string. For more information, see the [Technical documentation](#).

Managed identities

System-assigned managed identities have a lifecycle bound to the Azure resource. So the relocation strategy of the Azure resource determines how the system-assigned managed identity is handled. If a new instance of the resource is created in the target, then you have to grant the new system-assigned managed identity the same permissions as the system-assigned managed identity in source region.

On the other hand, user-assigned managed identity have an independent lifecycle, and you can map the user-assigned managed identity to the new resource in the target region. For more information, see [Managed identity overview](#).

Next steps

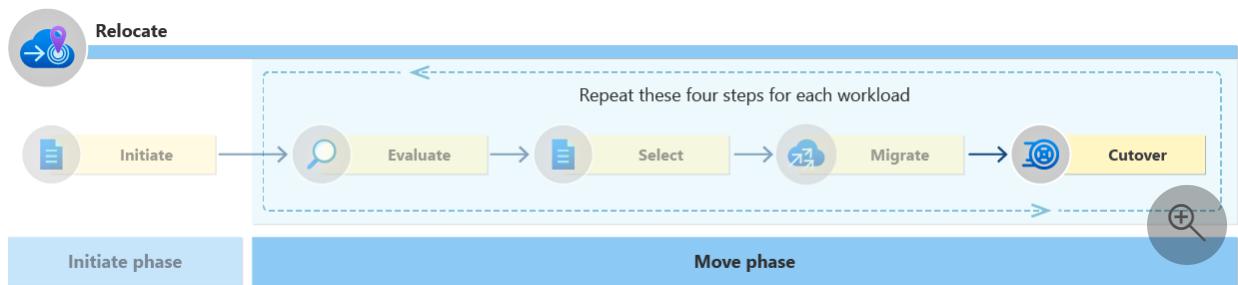
You migrated your workload services and data. Now you need to complete the relocation with a cutover.

Cutover

Cut over a cloud workload

Article • 12/19/2023

Cutover is when you direct traffic away from the source region and to the workload in the target region. After cutover, you can decommission the workload in the source region. To reduce costs and data deltas, you want the period between migration and cutover to be short. Here's the high-level process to cut over a cloud workload.



Test and validate migrated services and data. Test and validate the workload and dependencies to ensure the relocation was successful. Investigate and remediate bugs or issues related to scripts built by the relocation delivery team. You should run user acceptance tests (UATs). It's a best practice to assign different users to various parts of the application for UAT. You want to receive confirmation from users that the workload functions before switching the endpoints.

Switch endpoints. You should execute the cutover plan from the Select step of the relocation process. Have a failover strategy in place for urgent fixes.

Validate traffic. Validate the traffic is routed to the target region, for example, by running smoke tests. You should communicate the relocation progress to users so they're informed. Also, check the workload metrics and logs to confirmation the workload is working properly.

Fix if necessary. If something goes wrong, you should implement the failover plan or apply an urgent fix to stabilize the deployment.

Review operational configurations. Make sure you turn on or configure the new workload environments, including the updated artifacts (config files, wikis, readme), infrastructure as code (IaC), pipelines for your new environment. You should follow all Azure Advisor recommendations and configuring items such as backups, security controls, logging, and cost reporting.

Next steps

Repeat the Move phase for each workload. If you have more workloads to relocate, return to the [Evaluate step](#) and repeat the four steps of the Move phase until you complete the relocation project. Otherwise, you need to formally close the relocation project.

Close project. After you're done relocating, you should officially close the relocation project. Closure should take place two weeks after the final cutover. You need time to assess the success of the relocation and create a report for stakeholders to review. Business and technical stakeholders should review the report and approve.

Modernize workloads. Depending on the state of your workload, you might want to continue with our adopt guidance for modernizing workloads with Azure platform-as-a-service solutions (PaaS) or conduct a well-architected review to determine areas of improvement.

[Modernize workloads](#)

[Azure Well-Architected Framework](#)

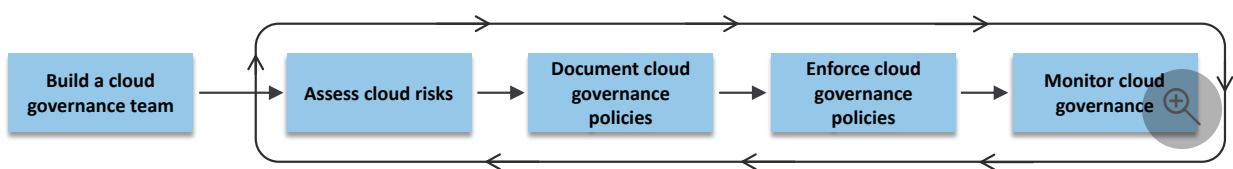
[Azure Well-Architected Review](#)

Govern overview

Article • 04/05/2024

The CAF Govern methodology provides a structured approach for establishing and optimizing cloud governance in Azure. The guidance is relevant for organizations across any industry. It covers essential categories of cloud governance, such as regulatory compliance, security, operations, cost, data, resource management, and artificial intelligence (AI).

Cloud governance is how you control cloud use across your organization. Cloud governance sets up guardrails that regulate cloud interactions. These guardrails are a framework of policies, procedures, and tools you use to establish control. Policies define acceptable and unacceptable cloud activity, and the procedures and tools you use ensure all cloud usage aligns with those policies. Successful cloud governance prevents all unauthorized or unmanaged cloud usage.



Why govern the cloud?

Cloud governance is foundational to defining and sustaining the productive use of the cloud. Effective cloud governance regulates all cloud use, mitigates risks, and streamlines cloud interactions across the organization. It aligns cloud use with the broader cloud strategy and helps you reach business goals with fewer setbacks. Without cloud governance, your organization might encounter risks that cloud governance could prevent.

How to govern the cloud?

Cloud governance is a continuous process. It requires ongoing monitoring, evaluation, and adjustments to adapt to evolving technologies, risks, and compliance requirements. The CAF Govern methodology divides cloud governance into five steps. Complete all five steps to establish cloud governance and regularly iterate on steps 2-5 to maintain cloud governance over time:

1. *Build a governance team:* Select a team of individuals to be responsible for cloud governance. The cloud governance team defines and maintains cloud governance

policies while reporting on the overall progress of cloud governance.

2. *Assess cloud risks:* Evaluate and prioritize potential risks associated with the use of the cloud. The risk assessment should identify risks unique to your organization. Consider all categories of risk, such as regulatory compliance, security, operations, cost, data, resource management, and AI risks. Use Azure tools to help [assess cloud risks](#).
3. *Document cloud governance policies:* Define the cloud governance policies that dictate the acceptable use of the cloud. These cloud governance policies set out the rules and guidelines for cloud usage to minimize the identified risks.
4. *Enforce cloud governance policies:* Enforce compliance with the cloud governance policies using automated tools or manual procedures. The goal is to ensure that the use of cloud services is in line with the established cloud governance policies. Use Azure tools to help [enforce cloud governance policies](#).
5. *Monitor cloud governance:* Monitor cloud use and teams responsible for governance to ensure they're compliant with the cloud governance policies. Use Azure tools to help [monitor cloud governance](#) and [set up alerts for noncompliance](#).

Cloud governance checklist

Use the cloud governance checklist to see all the tasks for each cloud governance step. Use the links to quickly navigate to the guidance you need.

[] [Expand table](#)

Cloud governance step	Cloud governance tasks
□ Build a cloud governance team.	□ Define the functions of the cloud governance team. □ Select the members of the cloud governance team. □ Define the authority of the cloud governance team. □ Define the scope of the cloud governance team.
□ Assess cloud risks.	□ Identify cloud risks. □ Analyze cloud risks. □ Document cloud risks. □ Communicate cloud risks. □ Review cloud risks.
□ Document cloud governance policies.	□ Define an approach for documenting cloud governance policies. □ Define cloud governance policies.

Cloud governance step	Cloud governance tasks
	<input type="checkbox"/> Distribute cloud governance policies. <input type="checkbox"/> Review cloud governance policies.
<input type="checkbox"/> Enforce cloud governance policies.	<input type="checkbox"/> Define an approach for enforcing cloud governance policies. <input type="checkbox"/> Enforce cloud governance policies automatically. <input type="checkbox"/> Enforce cloud governance policies manually. <input type="checkbox"/> Review policy enforcement.
<input type="checkbox"/> Monitor cloud governance.	<input type="checkbox"/> Configure cloud governance monitoring. <input type="checkbox"/> Configure cloud governance alerts. <input type="checkbox"/> Develop a remediation plan. <input type="checkbox"/> Audit cloud governance regularly.

Next step

[Build a cloud governance team](#)

Feedback

Was this page helpful?

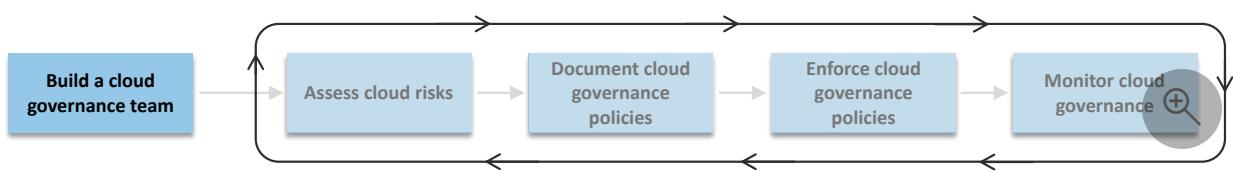
 Yes

 No

Build a cloud governance team

Article • 04/05/2024

This article shows you how to build a cloud governance team. A cloud governance team oversees cloud governance for the organization. This team is responsible for assessing risks, documenting cloud governance policies, and reporting on the progress of cloud governance. They need to understand the needs of teams across the business and ensure cloud governance policies minimize risks. The goal is to have people accountable for the success of cloud governance. To build a cloud governance team, complete these tasks.



Define the functions of the cloud governance team

Define the duties and roles of the cloud governance team. Outline the required functions and what they need to do to implement cloud governance effectively. At a minimum, the cloud governance team should fulfill the following functions:

- *Engage stakeholders.* The cloud governance team must actively engage stakeholders across the organization (IT, finance, operations, security, and compliance) to gather input on defining cloud governance policies. The goal is to ensure cloud governance policies minimize risk without preventing teams from achieving business goals.
- *Assess cloud risks.* The cloud governance team must oversee the identification, analysis, and prioritization of cloud risks. They oversee risk assessments and communicate findings to stakeholders. They provide access to tools to evaluate security, compliance, and operational cloud risks.
- *Develop and update governance policies.* The cloud governance team should document cloud governance policies for the organization. They should resolve any challenges that cloud governance creates for teams and should regularly review and update cloud governance policies as needed. The goal is to ensure the cloud governance policies are comprehensive, enforceable, and align with current technology and requirements.

- *Monitor and review governance.* Establish metrics to measure the effectiveness of cloud governance. Develop reporting methods to track compliance rates, incident response times, and user satisfaction.

Select the members of the cloud governance team

Select the individuals responsible for overseeing and managing cloud governance within the organization. Recruit members with the skills to efficiently enforce policies, manage risks, and comply with regulations. To select members of the cloud governance team, follow these recommendations:

- *Select a small team.* Pick a small team to encourage agility and quicker decision-making.
- *Select a diverse team.* The team should consist of individuals from different areas of the organization. Consider including IT operations, security, finance, software development, cloud architecture, and compliance.
- *Define team members' responsibilities.* Define the roles and responsibilities within your cloud governance team. Tailor them to your organization's size, complexity, and cloud maturity. Key areas of responsibility include the cloud governance success, cloud architecture, cloud security, cloud compliance, and cloud finance.

Define the authority of the cloud governance team

Empower the cloud governance team to implement and oversee cloud governance. The goal is to ensure the cloud governance team has the legitimacy and support required to achieve the organization's cloud governance objectives. To define the authority of the cloud governance team, follow these recommendations:

- *Secure executive sponsorship.* Gain support from and report to a named executive, such as the CIO or CTO, to support the cloud governance initiative. The executive sponsor serves as a point of escalation for challenges and helps align cloud governance with business goals.
- *Establish authority levels.* The executive sponsor should grant the team the authority to define cloud governance policies and take corrective measures for noncompliance.

- *Communicate authority.* The executive sponsor should communicate the authority of the cloud governance team to the entire organization. Include the importance of adhering to the cloud governance policies they create.

Define the scope of the cloud governance team

Establish the boundaries of the cloud governance team's responsibilities. The goal is to clarify areas of responsibility so the cloud governance team can focus on their defined functions. To define the scope, follow these recommendations:

- *Define relationship with other teams.* Clearly define the cloud governance team's authority over cloud resources, services, and policies. Avoid conflict and overlapping responsibility with other teams. For hybrid environments, specify the cloud governance team's responsibilities in contrast to on-premises teams.
- *Use a RACI matrix.* Use a responsibility assignment matrix, often called a RACI matrix, to delineate roles and responsibilities within the cloud governance framework.

Example cloud governance RACI matrix

The following table is an example of a RACI matrix for cloud governance. The matrix indicates who is responsible (R), accountable (A), consulted (C), and informed (I) across various cloud governance tasks. Create a RACI matrix that aligns to your organization and meets your specific needs.

[\[+\] Expand table](#)

Task	Cloud governance team	Executive sponsor	Cloud platform team	Workload teams
Engage stakeholders	R, A	I	C	C
Assess cloud risks	A	I	R	R
Develop and update governance policies	R, A	I	C	C
Report on cloud governance progress	R, A	I	C	C
Plan a cloud architecture	A	I	R	R
Enforce governance	A, C	I	R	R

Task	Cloud governance team	Executive sponsor	Cloud platform team	Workload teams
policies				
Monitor governance	A, C	I	R	R

Next step

[Assess cloud risks](#)

Feedback

Was this page helpful?

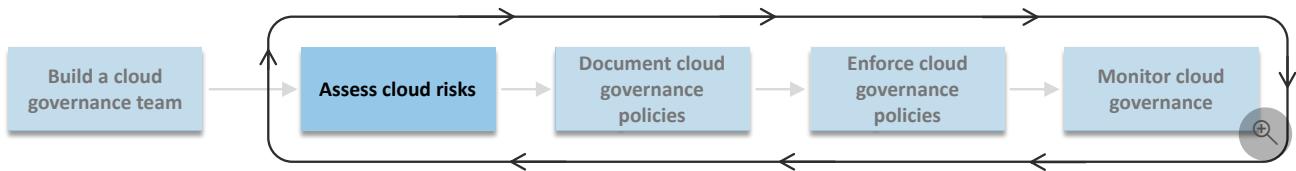
 Yes

 No

Assess cloud risks

Article • 04/05/2024

This article outlines how to assess risks associated with the cloud. All technologies introduce certain risks to an organization. Risks are undesired outcomes that could affect your business, such as noncompliance with industry standards. When adopting the cloud, you need to identify the risks the cloud poses to your organization. The cloud governance team creates cloud governance policies to prevent and mitigate those risks. To assess cloud risks, complete these tasks.



Identify cloud risks

Catalog a comprehensive list of cloud risks. Knowing your risks allows you to create cloud governance policies that can prevent and mitigate those risks. To identify cloud risks, follow these recommendations:

- *List all cloud assets.* List all your cloud assets so you can comprehensively identify the risks associated with them. For example, you can use the Azure portal, Azure Resource Graph, PowerShell, and Azure CLI to view all resources in a subscription.
- *Discover cloud risks.* Develop a stable risk catalog to guide cloud governance policies. To prevent frequent adjustments, focus on general cloud risks, not risks unique to a specific workload. Start with high-priority risks and develop a more comprehensive list over time. Common categories of risk are regulatory compliance, security, operations, cost, data, resources, and AI. Include risks that are unique to your organization, such as non-Microsoft software, partner or vendor support, and internal cloud competencies.
- *Involve key stakeholders.* Gather input from diverse organizational roles (IT, security, legal, finance, and business units) to consider all potential risks. This collaborative approach ensures a holistic view of risks related to the cloud.
- *Verify risks.* Engage external experts who possess a deep understanding of cloud risk identification to review and validate your risk list. These experts could be Microsoft account teams or specialized Microsoft partners. Their expertise helps confirm the identification of all potential risks and enhances the accuracy of your risk assessment.

Azure facilitation: Identifying cloud risks

The following guidance is meant to help you identify cloud risks in Azure. It provides a sample starting point for major categories of cloud governance. Azure can help automate part of the process of finding risks. Use Azure tools such as Azure Advisor, Microsoft Defender for Cloud, Azure Policy, Azure Service Health, and Microsoft Purview.

- *Identify regulatory compliance risks.* Identify risks of noncompliance with legal and regulatory frameworks affecting cloud data and operations. Know the regulatory requirements of your industry. Use the [Azure compliance documentation](#) to start.
- *Identify security risks.* Identify threats and vulnerabilities that jeopardize the confidentiality, integrity, and availability of the cloud environment. Use Azure to assess your [cloud security posture](#) and detect [identity risks](#).
- *Identify cost risks.* Identify risks related to the costs of cloud resources. Cost-related risks include overprovisioning, underutilization, and unexpected costs from data transfer fees or service scaling. Use a [cost assessment](#) to identify cost risk. Use Azure to estimate costs with the [Azure pricing calculator](#). [Analyze and forecast](#) costs on current resources. Identify [unexpected changes](#) in cloud costs.
- *Identify operations risks.* Identify risks that threaten the continuity of cloud operations, such as downtime and data loss. Use Azure tools to identify risks to [reliability and performance](#).
- *Identify data risks.* Identify risks related to data management within the cloud. Consider improper handling of data and flaws in data lifecycle management. Use Azure tools to help [identify data risks](#) and [explore risks to sensitive data](#).
- *Identify resource management risks.* Identify risks stemming from the provisioning, deployment, configuration, and management of cloud resources. Identify risks to [operational excellence](#).
- *Identify AI risks.* Regularly [red team language models](#). Manually test AI systems and supplement manual tests with [automated risk identification tools for AI](#). Look for common [human-AI interaction failures](#). Consider risks associated with use, access, and output of AI systems. Review the tenets of [responsible AI](#) and the [responsible AI maturity model](#).

Analyze cloud risks

Assign a qualitative or quantitative ranking to each risk so you can prioritize them by severity. Risk prioritization combines risk probability and risk impact. Prefer quantitative risk analysis over qualitative for more precise risk prioritization. To analyze cloud risks, follow these strategies:

Evaluate risk probability

Estimate the quantitative or qualitative probability of each risk occurring per year. Use a percentage range (0%-100%) to represent annual, quantitative risk probability. Low, medium, and high are common labels for qualitative risk probability. To evaluate risk probability, follow these recommendations:

- *Use public benchmarks.* Use data from reports, studies, or service-level agreements (SLAs) that document common risks and their occurrence rates.
- *Analyze historical data.* Look at internal incident reports, audit logs, and other records to identify how often similar risks occurred in the past.
- *Test control effectiveness.* To minimize risks, assess the effectiveness of current risk-mitigation controls. Consider reviewing control testing results, audit findings, and performance metrics.

Determine risk impact

Estimate the quantitative or qualitative impact of the risk occurring on the organization. A monetary amount is a common way to represent quantitative risk impact. Low, medium, and high are common labels for qualitative risk impact. To determine risk impact, follow these recommendations:

- *Conduct financial analysis.* Estimate the potential financial loss of a risk by looking at factors such as the cost of downtime, legal fees, fines, and the cost of remediation efforts.
- *Conduct reputational impact assessment.* Use surveys, market research, or historical data on similar incidents to estimate the potential impact on the organization's reputation.
- *Conduct operational disruption analysis.* Assess the extent of operational disruption by estimating downtime, loss of productivity, and the cost of alternative arrangements.
- *Assess legal implications.* Estimate potential legal costs, fines, and penalties associated with noncompliance or breaches.

Calculate risk priority

Assign a risk priority to each risk. Risk priority is the importance you assign to a risk so you know whether to treat the risk with high, medium, or low urgency. Risk impact is more important than risk probability since a high-impact risk can have lasting consequences. The governance team must use a consistent methodology across the organization to prioritize risk. To calculate risk priority, follow these recommendations:

- *Use a risk matrix for qualitative assessments.* Create a matrix to assign a qualitative risk priority to each risk. One axis of the matrix represents risk probability (high, medium, low) and the other represents risk impact (high, medium, low). The following table provides a sample risk matrix:

[+] Expand table

	Low impact	Medium impact	High impact
Low probability	Very low	Moderately low	Moderately high
Medium probability	Low	Medium	High
High probability	Medium	High	Very high

- *Use formulas for quantitative assessments.* Use the following calculation as a baseline: $risk\ priority = risk\ probability \times risk\ impact$. Adjust the weight of the variables as needed to tailor the risk priority results. For example, you could put more emphasis on the risk impact with this formula: $risk\ priority = risk\ probability \times (risk\ impact \times 1.5)$.

Assign a risk level

Categorize each risk into one of three levels: major risks (level 1), subrisks (level 2), and risk drivers (level 3). Risk levels allow you to plan an appropriate risk management strategy and anticipate future challenges. Level 1 risks threaten the organization or technology. Level 2 risks fall under the level 1 risk. Level 3 risks are trends that could potentially culminate in one or more level 1 or level 2 risks. For example, consider noncompliance with data protection laws (level 1), improper cloud storage configurations (level 2), and increasing complexity of regulatory requirements (level 3).

Determine risk management strategy

For each risk, identify appropriate risk treatment options, such as avoiding, mitigating, transferring, or accepting the risk. Provide an explanation of the choice. For example, if you decide to accept a risk because the cost of mitigating it's too expensive, you should document that reasoning for future reference.

Assign risk owners

Designate a primary risk owner for every risk. The risk owner has the responsibility of managing each risk. This person coordinates the risk management strategy across every team involved and is the initial point of contact for risk escalation.

Document cloud risks

Document each risk and the details of the risk analysis. Create a list of risks (risk register) that contains all the information you need to identify, categorize, prioritize, and manage risks. Develop standardized language for risk documentation so everyone can easily understand the cloud risks. Consider including these elements:

- *Risk ID*: A unique identifier for each risk. Increment the identifier sequentially as you add new risks. If you remove risks, you can leave gaps in the sequence or fill the gaps in the sequence.
- *Risk management status*: The status of the risk (open, closed).
- *Risk category*: A label such as regulatory compliance, security, cost, operations, AI, or resource management.
- *Risk description*: A brief description of the risk.
- *Risk probability*: The probability of the risk occurring per year. Use a percentage or qualitative label.
- *Risk impact*: The impact on the organization if the risk occurs. Use a monetary amount or qualitative label.
- *Risk priority*: The severity of the risk (probability x impact). Use a dollar amount or qualitative label.
- *Risk level*: The type of risk. Use major threat (level 1), subrisk (level 2), or risk driver (level 3).
- *Risk management strategy*: The approach to manage the risk such as mitigate, accept, or avoid.
- *Risk management enforcement*: The techniques to enforce the risk management strategy.
- *Risk owner*: The individual managing the risk.
- *Risk closure date*: A date when the risk management strategy should be applied.

For more information, see [Risk list example](#).

Communicate cloud risks

Clearly convey identified cloud risks to the executive sponsor and executive-level management. The goal is to ensure the organization prioritizes cloud risks. Provide regular updates on cloud risk management and communicate when you need extra resources to manage risks. Promote a culture where managing cloud risks management and governance is a part of daily operations.

Review cloud risks

Review the current cloud risk list to ensure it's valid and accurate. Reviews should be regular and also in response to specific events.

Maintain, update, or remove risks as needed. To review cloud risks, follow these recommendations:

- *Schedule regular assessments*. Set a recurring schedule to review and assess cloud risks, such as quarterly, biannually, or yearly. Find a review frequency that best accommodates personnel availability, the rate of cloud environment changes, and organizational risk tolerance.
- *Conduct event-based reviews*. Review risks in response to specific events, such as the failed prevention of a risk. Consider reviewing risks when you adopt new technologies, change business processes, and discover new security threats events. Also consider reviewing when technology, regulatory compliance, and organizational risk tolerance changes.
- *Review cloud governance policies*. Keep, update, or remove cloud governance policies to address new risks, existing risks, or outdated risks. Review the cloud governance policy statement and cloud governance enforcement strategy as needed. When you remove a risk,

evaluate if the cloud governance policies associated with it are still relevant. Consult with stakeholders to remove the cloud governance policies or update the policies to associate them with a new risk.

Example risk list

The following table is an example risk list, also known as a risk register. Tailor the example to fit the specific needs and context of your organization's Azure cloud environment.

[Expand table](#)

Risk ID	Risk management status	Risk category	Risk description	Risk probability	Risk impact	Risk priority	Risk level	Risk management strategy	Risk management enforcement	Risk owner	Risk closure date
R01	Open	Regulatory compliance	Noncompliance with sensitive data requirements	20% OR Medium	\$100,000 OR High	\$20,000 OR High	Level 2	Mitigate	Use Microsoft Purview for sensitive data monitoring. Compliance reporting in Microsoft Purview.	Compliance lead	2024-04-01
R02	Open	Security	Unauthorized access to cloud services	30% OR High	\$200,000 OR High	\$60,000 OR Very high	Level 1	Mitigate	Microsoft Entra ID multifactor authentication (MFA). Microsoft Entra ID Governance monthly access reviews.	Security lead	2024-03-15
R03	Open	Security	Insecure code management	20% OR Medium	\$150,000 OR High	\$30,000 OR High	Level 2	Mitigate	Use defined code repository. Use quarantine pattern for public libraries.	Developer lead	2024-03-30
R04	Open	Cost	Overspending on cloud services due to overprovisioning and lack of monitoring	40% OR High	\$50,000 OR Medium	\$20,000 OR High	Level 2	Mitigate	Set budgets and alerts for workloads. Review and apply Advisor cost recommendations.	Cost lead	2024-03-01
R05	Open	Operations	Service disruption due to Azure region outage	25% OR Medium	\$150,000 OR High	\$37,500 OR High	Level 1	Mitigate	Mission-critical workloads have active-active architecture. Other workloads have active-passive architecture.	Operations lead	2024-03-20
R06	Open	Data	Loss of sensitive data due to improper encryption and data lifecycle management	35% OR High	\$250,000 OR High	\$87,500 OR Very high	Level 1	Mitigate	Apply encryption in transit and at rest. Establish data lifecycle policies using Azure tools.	Data lead	2024-04-10
R07	Open	Resource management	Misconfiguration of cloud resources leading to unauthorized access and data exposure	30% OR High	\$100,000 OR High	\$30,000 OR Very high	Level 2	Mitigate	Use infrastructure as code (IaC). Enforce tagging requirements using Azure Policy.	Resource lead	2024-03-25
R08	Open	AI	AI model producing biased decisions due to	15% OR Low	\$200,000 OR High	\$30,000 OR Moderately high	Level 3	Mitigate	Use content filtering mitigation techniques. Red team AI models monthly.	AI lead	2024-05-01

Risk ID	Risk management status	Risk category	Risk description	Risk probability	Risk impact	Risk priority	Risk level	Risk management strategy	Risk management enforcement	Risk owner	Risk closure date
			unrepresentative training data								

Next step

[Document cloud governance policies](#)

Feedback

Was this page helpful?

 Yes

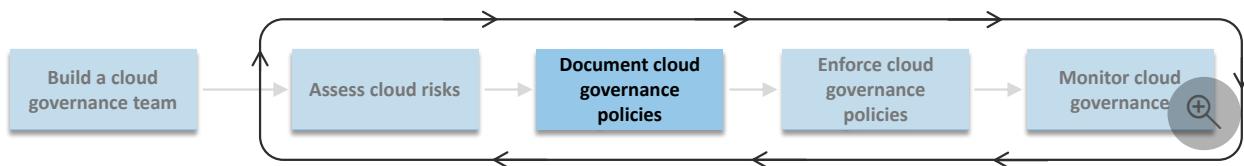
 No

Document cloud governance policies

Article • 04/05/2024

This article shows you how to define and document cloud governance policies. Cloud governance policies specify what should or shouldn't happen in the cloud. The cloud governance team should create one or more cloud governance policies for each risk identified in the risk assessment.

Cloud governance policies define the guardrails for interacting with and in the cloud.



Define an approach for documenting cloud governance policies

Establish an approach for creating, maintaining, and updating the rules and guidelines that govern the use of cloud services. Cloud governance policies shouldn't be unique to a specific workload. The goal is to produce cloud governance policies that don't require frequent updates and that consider the effects of cloud governance policies across the cloud environment. To define a policy documentation approach, follow these recommendations:

- *Define standard governance language.* Develop a standard structure and format for documenting cloud governance policies. The policies must be a clear and authoritative reference for stakeholders.
- *Recognize the different scopes of governance.* Define and assign specific governance responsibilities tailored to the unique roles within your organization. For example, a developer governs application code. A workload team is responsible for a single workload, and the platform team is responsible for governance that workloads inherit.
- *Evaluate the broad effects of cloud governance.* Cloud governance creates friction. Find a balance between friction and freedom. Consider the effects of governance on workload architecture, software development practices, and other areas as you develop cloud governance policies. For example, what you allow or disallow determines workload architecture and affects software development practices.

Define cloud governance policies

Create cloud governance policies that outline how to use and manage the cloud to mitigate risks. Minimize the need for frequent policy updates. To define cloud governance policies, follow these recommendations:

- *Use a policy ID.* Use the policy category and a number to uniquely identify each policy, such as *SC01* for the first security governance policy. Increment the identifier sequentially as you add new risks. If you remove risks, you can leave gaps in the sequence or use the lowest available number.
- *Include the policy statement.* Craft specific policy statements that address identified risks. Use definitive language such as *must*, *should*, *must not*, and *shouldn't*. Use the enforcement controls from the risk list as a starting point. Focus on outcomes rather than configuration steps. Name the tool required for enforcement so you know where to monitor compliance.
- *Include a risk ID.* List the risk in the policy. Associate every cloud governance policy to a risk.
- *Include the policy category.* Include governance categories, such as security, compliance, or cost management, into the policy categorization. Categories help with sorting, filtering, and finding cloud governance policies.
- *Include the policy purpose.* State the purpose of each policy. Use the risk or the regulatory compliance requirement the policy satisfies as a starting point.
- *Define the policy scope.* Define what and who this policy applies to, such as all cloud services, regions, environments, and workloads. Specify any exceptions to ensure there's no ambiguity. Use standardized language so it's easy to sort, filter, and find policies.
- *Include the policy remediation strategies.* Define the desired response to a violation of a cloud governance policy. Tailor responses to the severity of the risk, such as scheduling discussions for nonproduction violations and immediate remediation efforts for production violations.

For more information, see the [example cloud governance policies](#).

Distribute cloud governance policies

Grant access to everyone who needs to adhere to cloud governance policies. Look for ways to make adherence to the cloud governance policies easier for people in your organization. To distribute cloud governance policies, follow these recommendations:

- *Use a centralized policy repository.* Use a centralized, easily accessible repository for all governance documentation. Ensure all stakeholders, teams, and individuals have access to the latest versions of policies and related documents.
- *Create compliance checklists.* Provide a quick and actionable overview of the policies. Make it easier for teams to comply without having to navigate through extensive documentation.
For more information, see the [example compliance checklist](#).

Review cloud governance policies

Assess and update cloud governance policies to ensure they remain relevant and effective in governing cloud environments. Regular reviews help ensure that cloud governance policies align with changing regulatory requirements, new technologies, and evolving business objectives.

When you review policies, consider the following recommendations:

- *Implement feedback mechanisms.* Establish ways to receive feedback on the effectiveness of cloud governance policies. Gather input from the individuals affected by the policies to ensure they can still do their job efficiently. Update governance policies to reflect practical challenges and needs.
- *Establish event-based reviews.* Review and update cloud governance policies in response to events, such as a failed governance policy, technology change, or regulatory compliance change.
- *Schedule regular reviews.* Regularly review governance policies to ensure they align with evolving organizational needs, risks, and cloud advancements. For example, include governance reviews in the regular cloud governance meetings with stakeholders.
- *Facilitate change control.* Include a process for policy review and updates. Ensure the cloud governance policies stay aligned with organizational, regulatory, and technological changes. Make it clear how to edit, remove, or add policies.
- *Identify inefficiencies.* Review governance policies to find and fix inefficiencies in cloud architecture and operations. For example, instead of mandating that each workload must use its own web application firewall, update the policy to require the use of a centralized firewall. Review policies that require duplicated effort and see if there's a way to centralize the work.

Example cloud governance policies

The following cloud governance policies are examples for reference. These policies are based on the examples in the [example risk list](#).

[\[+\] Expand table](#)

Policy ID	Policy category	Risk ID	Policy statement	Purpose	Scope	Remediation	Monitoring
RC01	Regulatory compliance	R01	Microsoft Purview must be used to monitor sensitive data.	Regulatory compliance	Workload teams, platform team	Immediate action by affected team, compliance training	Microsoft Purview
RC02	Regulatory compliance	R01	Daily sensitive data compliance reports must be generated from	Regulatory compliance	Workload teams, platform team	Resolution within one day, confirmation audit	Microsoft Purview

Policy ID	Policy category	Risk ID	Policy statement	Purpose	Scope	Remediation	Monitoring
			Microsoft Purview.				
SC01	Security	R02	Multifactor authentication (MFA) must be enabled for all users.	Mitigate data breaches and unauthorized access	Azure users	Revoke user access	Microsoft Entra ID Conditional Access
SC02	Security	R02	Access reviews must be conducted monthly in Microsoft Entra ID Governance.	Ensure data and service integrity	Azure users	Immediate access revocation for noncompliance	ID Governance
SC03	Security	R03	Teams must use the specified GitHub organization for secure hosting of all software and infrastructure code.	Ensure secure and centralized management of code repositories	Development teams	Transfer of unauthorized repositories to the specified GitHub organization and potential disciplinary actions for noncompliance	GitHub audit log
SC04	Security	R03	Teams that use libraries from public sources must adopt the quarantine pattern.	Ensure libraries are safe and compliant before integration into the development process	Development teams	Removal of noncompliant libraries and review of integration practices for affected projects	Manual audit (monthly)
CM01	Cost management	R04	Workload teams must set budgets alerts at the resource group level.	Prevent overspending	Workload teams, platform team	Immediate reviews, adjustments for alerts	Microsoft Cost Management
CM02	Cost management	R04	Azure Advisor cost recommendations must be reviewed.	Optimize cloud usage	Workload teams, platform team	Mandatory optimization audits after 60 days	Advisor
OP01	Operations	R05	Production workloads should have an active-passive	Ensure service continuity	Workload teams	Architecture evaluations, biannual reviews	Manual audit (per production release)

Policy ID	Policy category	Risk ID	Policy statement	Purpose	Scope	Remediation	Monitoring
OP02	Operations	R05	All mission-critical workloads must implement a cross-region active-active architecture.	Ensure service continuity	Mission-critical workload teams	Updates within 90 days, progress reviews	Manual audit (per production release)
DG01	Data	R06	Encryption in transit and at rest must be applied to all sensitive data.	Protect sensitive data	Workload teams	Immediate encryption enforcement and security training	Azure Policy
DG02	Data	R06	Data lifecycle policies must be enabled in Microsoft Purview for all sensitive data.	Manage the data lifecycle	Workload teams	Implementation within 60 days, quarterly audits	Microsoft Purview
RM01	Resource management	R07	Bicep must be used to deploy resources.	Standardize resource provisioning	Workload teams, platform team	Immediate Bicep transition plan	Continuous integration and continuous delivery (CI/CD) pipeline
RM02	Resource management	R07	Tags must be enforced on all cloud resources using Azure Policy.	Facilitate resource tracking	All cloud resources	Correct tagging within 30 days	Azure Policy
AI01	AI	R08	AI content filtering configuration must be set to medium or higher.	Mitigate AI harmful outputs	Workload teams	Immediate corrective measures	Azure OpenAI Service
AI02	AI	R08	Customer-facing AI systems must be red-teamed monthly.	Identify AI biases	AI model teams	Immediate review, corrective actions for misses	Manual audit (monthly)

Next step

Enforce cloud governance policies

Feedback

Was this page helpful?

 Yes

 No

Enforce cloud governance policies

Article • 04/17/2024

This article shows you how to enforce compliance with cloud governance policies. Cloud governance enforcement refers to the controls and procedures you use to align cloud use to the cloud governance policies. The cloud governance team assesses cloud risks and creates cloud governance policies to manage those risks. To ensure compliance with the cloud governance policies, the cloud governance team must delegate enforcement responsibilities. They must empower each team or individual to enforce cloud governance policies within their area of responsibility. The cloud governance team can't do it all. Prefer automated enforcement controls but enforce compliance manually where you can't automate.



Define an approach for enforcing cloud governance policies

Establish a systematic strategy to enforce compliance with cloud governance policies. The goal is to use automated tools and manual oversight to enforce compliance efficiently. To define an enforcement approach, follow these recommendations:

- *Delegate governance responsibilities.* Empower individuals and teams to enforce governance within their scope of responsibility. For example, platform teams should apply policies that the workloads inherit and workload teams should enforce governance for their workload. The cloud governance team shouldn't be responsible for applying enforcement controls.
- *Adopt an inheritance model.* Apply a hierarchical governance model where specific workloads inherit governance policies from the platform. This model helps ensure that organizational standards apply to the correct environments, such as purchasing requirements for cloud services. Follow the [design principles](#) of Azure landing zones and its [resource organization](#) design area to establish a proper inheritance model.
- *Discuss enforcement specifics.* Discuss where and how you apply governance policies. The goal is to find cost effective ways to enforce compliance that accelerates productivity. Without a discussion, you risk blocking the progress of

specific teams. It's important to find a balance that supports the business objectives while managing risk effectively.

- *Have a monitor-first stance.* Don't block actions unless you understand them first. For lower priority risk, start by monitoring compliance with cloud governance policies. After you understand the risk, you can move to more restrictive enforcement controls. A monitor-first approach gives you an opportunity to discuss the governance needs and realign the cloud governance policy and enforcement control to those needs.
- *Prefer blocklists.* Prefer blocklists over allowlists. Blocklists prevent the deployment of specific services. It's better to have a small list of services that you shouldn't use than a long list of services you can use. To avoid lengthy blocklists, don't add new services to the blocklist by default.
- *Define a tagging and naming strategy.* Establish systematic guidelines for naming and tagging cloud resources. It provides a structured framework for resource categorization, cost management, security, and compliance across the cloud environment. Allow teams, such as development teams, to add other tags for their unique needs.

Enforce cloud governance policies automatically

Use cloud management and governance tools to automate compliance with governance policies. These tools can help in setting up guardrails, monitoring configurations, and ensuring compliance. To set up automated enforcement, follow these recommendations:

- *Start with a small set of automated policies.* Automate compliance on a small set of essential cloud governance policies. Implement and test automation to avoid operational disruptions. Expand your list of automated enforcement controls when ready.
- *Use cloud governance tools.* Use the tools available in your cloud environment to enforce compliance. Azure's primary governance tool is [Azure Policy](#). Supplement Azure Policy with [Microsoft Defender for Cloud](#) (security), [Microsoft Purview](#) (data), [Microsoft Entra ID Governance](#) (identity), [Azure Monitor](#) (operations), [management groups](#) (resource management), [infrastructure as code \(IaC\)](#) (resource management), and configurations within each Azure service.
- *Apply governance policies at the right scope.* Use an inheritance system where policies are set at a higher level, such as management groups. Policies at higher

levels automatically apply to lower levels, such as subscriptions and resource groups. Policies apply even when there are changes within the cloud environment, lowering management overhead.

- *Use policy enforcement points.* Set up policy enforcement points within your cloud environments that automatically apply governance rules. Consider predeployment checks, runtime monitoring, and automated remediation actions.
- *Use policy as code.* Use IaC tools to [enforce governance policies through code](#). Policy as code enhances the automation of governance controls and ensures consistency across different environments. Consider using [Enterprise Azure Policy as Code](#) (EPAC) to manage policies aligned with recommended Azure landing zone policies.
- *Develop custom solutions as needed.* For custom governance actions, consider developing custom scripts or applications. Use Azure service APIs to gather data or manage resources directly.

Azure facilitation: Enforcing cloud governance policies automatically

The following guidance can help you find the right tools to automate compliance with cloud governance policies in Azure. It provides a sample starting point for major categories of cloud governance.

Automate regulatory compliance governance

- *Apply regulatory compliance policies.* Use [built-in regulatory compliance policies](#) that align with compliance standards, such as HITRUST/HIPAA, ISO 27001, CMMC, FedRamp, and PCI DSSv4.
- *Automate custom restrictions.* [Create custom policies](#) to define your own rules for working with Azure.

Automate security governance

- *Apply security policies.* Use the [built-in security policies](#) and [automated security compliance](#) to align with common security standards. There's built-in policies for NIST 800 SP series, Center for Internet Security benchmarks, and the Microsoft cloud security benchmark. Use built-in policies to [automate the security configuration](#) of specific Azure services. [Create custom policies](#) to define your own rules for working with Azure.

- *Apply identity governance.* Enable [Microsoft Entra multifactor authentication \(MFA\)](#) and [self-service password reset](#). [Eliminate weak passwords](#). Automate other aspects of [identity governance](#), such as access request workflows, access reviews, and identity lifecycle management. [Enable just-in-time access](#) to limit access to important resources. Use [conditional access](#) policies to [grant or block](#) user and [device identities](#) access to cloud services.
- *Apply access controls.* Use [Azure role-based access control \(RBAC\)](#) and [attribute-based access control](#) (ABAC) to govern access to specific resources. Grant and deny permissions to users and groups. Apply the permission at the appropriate [scope](#) (management group, subscription, resource group, or resource) to provide only the permission needed and limit management overhead.

Automate cost governance

- *Automate deployment restrictions.* [Disallow certain cloud resources](#) to prevent the use of cost-intensive resources.
- *Automate custom restrictions.* [Create custom policies](#) to define your own rules for working with Azure.
- *Automate cost allocation.* Enforce tagging requirements to [group and allocate costs](#) across environments (development, test, production), departments, or projects. Use tags to identify and track resources that are part of a cost optimization effort.

Automate operations governance

- *Automate redundancy.* Use built-in Azure policies to require a specified level of infrastructure redundancy, such as zone-redundant and geo-redundant instances.
- *Apply backup policies.* Use [backup policies](#) to govern the backup frequency, retention period, and storage location. Align backups policies with data governance, regulatory compliance requirements, recovery time objective (RTO), and recovery point objective (RPO). Use the backup settings in individual Azure services, such as [Azure SQL Database](#), to configure the settings you need.
- *Meet the target service-level objective.* Restrict the deployment of certain services and service tiers (SKUs) that don't meet your target service-level objective. For example, use the `Not allowed resource types` policy definition in Azure Policy.

Automate data governance

- *Automate data governance.* Automate [data governance](#) tasks, such as cataloging, mapping, securely sharing, and applying policies.
- *Automate data lifecycle management.* Implement storage policies and [lifecycle management for storage](#) to ensure data is stored efficiently and compliantly.
- *Automate data security.* Review and enforce [data protection strategies](#), such as data segregation, encryption, and redundancy.

Automate resource management governance

- *Create a resource management hierarchy.* Use [management groups](#) to organize your subscriptions so that you can efficiently govern policies, access, and spending. Follow Azure landing zone [resource organization](#) best practices.
- *Enforce a tagging strategy.* Ensure all Azure resources are consistently tagged to improve manageability, cost tracking, and compliance. [Define your tagging strategy](#) and [manage tag governance](#).
- *Restrict which resources you can deploy.* [Disallow resource types](#) to restrict deployments of services that add unnecessary risk.
- *Restrict deployments to specific regions.* Control where resources are deployed to comply with regulatory requirements, manage costs, and reduce latency. For example, use the [Allowed locations](#) policy definition in Azure Policy. Also [enforce regional restrictions](#) in your deployment pipeline.
- *Use infrastructure as code (IaC).* Automate infrastructure deployments using [Bicep](#), [Terraform](#), or [Azure Resource Manager templates \(ARM templates\)](#). Store your IaC configurations in a source control system (GitHub or Azure Repos) to track changes and collaborate. Use [Azure landing zone accelerators](#) to govern the deployment of your platform and application resources and avoid configuration drift over time.
- *Govern hybrid and multicloud environments.* [Govern hybrid and multicloud resources](#). Maintain consistency in management and policy enforcement.

Automate AI governance

- *Use the retrieval augmented generation (RAG) pattern.* RAG adds an information retrieval system to control the grounding data that a language model uses to generate a response. For example, you can use [the Azure OpenAI Service](#) on your

own data feature or set up RAG with [Azure AI Search](#) to constrain generative AI to your content.

- *Use AI development tools.* Use AI tools, like Semantic Kernel, that facilitate and standardize AI orchestration when developing applications that use AI.
- *Govern output generation.* Help prevent abuse and harmful content generation. Use [AI content filtering](#) and [AI abuse monitoring](#).
- *Configure data loss prevention.* Configure [data loss prevention for Azure AI services](#). Configure the list of outbound URLs that their AI services resources are allowed to access.
- *Use system messages.* Use [system messages](#) to guide the behavior of an AI system and tailor the outputs.
- *Apply the AI security baseline.* Use the [Azure AI security baseline](#) to govern the security of AI systems.

Enforce cloud governance policies manually

Sometimes a tool limitation or cost makes automated enforcement unpractical. In cases where you can't automate enforcement, enforce cloud governance policies manually. To manually enforce cloud governance, follow these recommendations:

- *Use checklists.* Use governance checklists to make it easy for your teams to follow the cloud governance policies. For more information, see the [example compliance checklists](#).
- *Provide regular training.* Conduct frequent training sessions for all relevant team members to ensure they're aware of the governance policies.
- *Schedule regular reviews.* Implement a schedule for regular reviews and audits of cloud resources and processes to ensure compliance with governance policies. These reviews are critical for identifying deviations from established policies and taking corrective actions.
- *Monitor manually.* Assign dedicated personnel to monitor the cloud environment for compliance with governance policies. Consider tracking the use of resources, managing access controls, and ensuring data protection measures are in place to align with the policies. For example, define a [comprehensive cost management approach](#) to govern cloud costs.

Review policy enforcement

Regularly review and update compliance enforcement mechanisms. The goal is to keep cloud governance policy enforcement aligned with current needs, including developer, architect, workload, platform, and business requirements. To review policy enforcement, follow these recommendations:

- *Engage with stakeholders.* Discuss the effectiveness of enforcement mechanisms with stakeholders. Ensure cloud governance enforcement aligns with business objectives and compliance requirements.
- *Monitor requirements.* Update or remove enforcement mechanisms to align with new or updated requirements. Track changes in regulations and standards that require updates your enforcement mechanisms. For example, Azure landing zone recommended policies can change over time. You should [detect](#) those policy changes, [update](#) to the latest Azure landing zone custom policies, or [migrate](#) to built-in policies as needed.

Example cloud governance compliance checklists

Compliance checklists help teams understand the governance policies that apply to them. The example compliance checklists use the policy statement from the [example cloud governance policies](#) and contain the cloud governance policy ID for cross-referencing.

[] Expand table

Category	Compliance requirement
Regulatory compliance	<input type="checkbox"/> Microsoft Purview must be used to monitor sensitive data (RC01). <input type="checkbox"/> Daily sensitive data compliance reports must be generated from Microsoft Purview (RC02).
Security	<input type="checkbox"/> MFA must be enabled for all users (SC01). <input type="checkbox"/> Access reviews must be conducted monthly in ID Governance (SC02). <input type="checkbox"/> Use the specified GitHub organization to host all application and infrastructure code (SC03). <input type="checkbox"/> Teams that use libraries from public sources must adopt the quarantine pattern (SC04).
Operations	<input type="checkbox"/> Production workloads should have an active-passive architecture across regions (OP01).

Category	Compliance requirement
	<input type="checkbox"/> All mission-critical workloads must implement a cross-region active-active architecture (OP02).
Cost	<input type="checkbox"/> Workload teams must set budgets alerts at the resource group level (CM01). <input type="checkbox"/> Azure Advisor cost recommendations must be reviewed (CM02).
Data	<input type="checkbox"/> Encryption in transit and at rest must be applied to all sensitive data. (DG01) <input type="checkbox"/> Data lifecycle policies must be enabled for all sensitive data (DG02).
Resource management	<input type="checkbox"/> Bicep must be used to deploy resources (RM01). <input type="checkbox"/> Tags must be enforced on all cloud resources using Azure Policy (RM02).
AI	<input type="checkbox"/> The AI content filtering configuration must be set to medium or higher (AI01). <input type="checkbox"/> Customer-facing AI systems must be red-teamed monthly (AI02).

Next step

[Monitor cloud governance](#)

Feedback

Was this page helpful?

 Yes

 No

Monitor cloud governance

Article • 04/05/2024

This article shows you how to monitor cloud governance. After you enforce cloud governance, you need to measure how aligned (compliant) your cloud environment is with your cloud governance policies. Start by taking an initial compliance measurement to identify areas that require improvement in order to align your cloud setup with your governance policies. Track compliance over time to see where cloud governance is effective and ineffective. The goal is to monitor governance and reduce noncompliance problems to zero.



Configure cloud governance monitoring

Implement monitoring solutions to track compliance with your cloud governance policies. The goal is to have visibility on the teams responsible for enforcing compliance so you can remediate noncompliance quickly. To configure governance monitoring, follow these recommendations:

- *Use monitoring tools.* Choose compliance monitoring tools that offer real-time monitoring capabilities. Ensure they can monitor compliance with your specific governance policies. Collect the metrics and logs as required for governance monitoring. Review the [visibility](#) and [monitoring](#) recommendations in the Azure landing zone management design area.
- *Manually monitor where necessary.* Review compliance manually where automated monitoring mechanisms aren't available.
- *Document monitoring solution.* Track how you're monitoring each cloud governance policy so you know where to gather compliance data. In the cloud governance policy, list the monitoring tool, such as Azure Policy or Microsoft Purview. If there's a manual approach, list the audit frequency.
- *Centralize governance monitoring.* Use or build a solution that allows you to view the status of cloud governance compliance in one place. For example, the [Azure governance workbook](#) centralizes many Azure governance monitoring services.

- *Establish a compliance baseline.* Evaluate how compliant your cloud environment is to your cloud governance policies. Make that your baseline. Track progress against the baseline over time.
- *Provide access to governance monitoring.* Configure the appropriate level of access to governance monitoring results so the teams responsible for governance can evaluate the effectiveness of enforcement controls.
- *Audit monitoring effectiveness.* Manually review compliance to validate compliancy. For example, ensure tags are receiving the right values and not an undesired value, such as NA.

Azure facilitation: Configuring cloud governance monitoring

The following guidance is meant to help you configure cloud governance monitoring in Azure. It provides a sample starting point for major categories of cloud governance. Consider aggregating these signals in the [Azure governance workbook](#). To configure cloud governance monitoring, you need an Azure identity that has permissions to gather monitoring data from your subscriptions.

Configure monitoring for regulatory compliance governance

- *Use compliance dashboards.* [Get policy compliance data](#) on the policies you assigned.
- *Determine compliance.* Use the compliance data to [determine causes of noncompliance](#).

Configure monitoring for security governance

- *Use security governance monitoring.* [Review security recommendations](#) and monitor security governance over time with your [secure score](#). The feature provides a dashboard to monitor [regulatory compliance](#) against common security frameworks.
- *Configure identity governance monitoring.* [Configure identity monitoring](#) to collect audit, sign-in, and provisioning logs. Also review your [identity secure score](#), and use [the identity governance dashboard](#) to get a single view of identities across your tenant.

Configure monitoring for cost management governance

- *Analyze cloud costs.* Conduct a [cost analysis](#) in Azure to gain full visibility into your cloud costs.
- *Create budgets.* [Create a budget](#) that aligns with your desired investment in the cloud.
- *Gather cost data.* Use [cost optimization recommendations](#) and the [cost optimization workbook](#) to guide cost management efforts, such as detection of idle resources. Identify anomalies and [unexpected changes in cost](#).

Configure monitoring for operations governance

- *Monitor policies on cloud operations.* Use Azure Policy to track compliance with governance policies that apply to operations.
- *Monitor logs and metrics.* To track [availability](#) and [performance](#), analyze [logs](#) and [metrics](#) across cloud environments.
- *Monitor resource optimization.* Use [Azure Advisor](#) to monitor Azure resources for reliability, security, operational excellence, performance, and cost. [Set alerts](#) for any new Advisor recommendations.
- *Monitor resource health.* [Monitor the health of Azure services](#) and monitor service-impacting events, planned maintenance, and other changes that might affect availability.

Configure monitoring for data governance

- *Monitor data governance.* [Monitor data compliance, management, and usage](#).
- *Use dashboards.* Use dashboards to monitor compliance with any data plane policies.

Configure monitoring for resource management governance

- *Monitor policies on resource management.* Monitor compliance with cloud governance policies that apply to resource deployments, such as tag enforcement policies.

Configure monitoring for AI governance

- *Monitor AI system outputs.* Use Azure for [abuse monitoring](#) and [content filtering](#) of AI systems.
- *Red team AI systems.* Regularly [red team language models](#) to find harmful outputs. Use both manual tests and automated tools to review the risk baseline.

Configure cloud governance alerts

Configure alerts based on specific compliance metrics or events that indicate a deviation from your governance policies. To configure cloud governance alerts, follow these recommendations:

- *Use cloud-native alerting mechanisms.* Prefer cloud-native tools that provide real-time monitoring and alerts for compliance problems.
- *Define noncompliance.* Define clear thresholds and baselines for noncompliance. Set alerts when data exceeds these thresholds or when unexpected changes occur that could indicate noncompliance.
- *Route alerts appropriately.* Send alerts to the appropriate team or individual responsible for enforcing compliance with cloud governance policies.
- *Include noncompliance information in alerts.* Configure alerts to include detailed information about the noncompliance event. Ideally include the policy violated, affected resources, and suggested remediation.

Azure facilitation: Configuring cloud governance alerts

The following guidance helps you start configuring cloud governance alerts in Azure. It provides a sample starting point for major categories of cloud governance.

- *Regulatory compliance governance alerts.* Use Azure activity logs to [generate alerts](#) for noncompliance across Azure.
- *Security governance alerts.* Configure [security alerts](#) and noncompliance [alerts](#).
- *Cost governance alerts.* Set up alerts to notify teams of potential cost overruns and spending anomalies. Configure [cost alerts](#) and [cost anomaly alerts](#). Set [reservation utilization alerts](#) to keep reservations and savings plans usage at or close to full usage.
- *Operations governance alerts.* [Configure alerts](#) on specific logs and metrics. [Set alerts](#) for new recommendations aligned to reliability and performance. Configure

service health alerts to get notified of current and upcoming service health problems. Configure [resource health alerts](#) to get notified of the current and historical health status of your Azure resources.

- *Data governance alerts.* Configure [data governance alerts](#) to report data governance violations.
- *Resource management governance alerts.* Configure alerts for when a noncompliance resource deploys. For example, use build warnings in your deployment pipeline or monitor noncompliance states.
- *AI governance alerts.* Configure alerts when there are harmful inputs and outputs in your AI systems. For example, monitor emails from Azure OpenAI that notify you of abusive behavior.

Develop a remediation plan

Develop a targeted action plan to address any noncompliance events. When you detect noncompliance, perform the remediation plan to correct the deviations and minimize the risk and impact. Add the remediation details to the cloud governance policy for easy access. Follow these recommendations:

- *Discuss remediation timeline.* Negotiate a timeline for remediation depending on risk priority. The team responsible for compliance must remediate compliance in a timely manner.
- *Remediate high-risk violations quickly.* For noncompliance alerts that are high risk, such as an exposed data endpoint, have a plan to escalate and fix those noncompliance problems. Update the policy enforcement mechanism to avoid a repeat of this high-risk violation. Use Azure to [react to compliance-state changes](#), [remediate resources noncompliant with policies](#), and [remediate security recommendations](#).
- *Follow up on low-risk violations.* Have an audit-first stance on low-risk policies so that you can have a discussion with the team that violated the cloud governance policy, such as deploying a service on a blocklist. Maybe there's a new feature available, better service tier (SKU), or a better price in a specific region. The cloud governance team should discuss the needs of the team and adjust policies and enforcement mechanisms in accordance with the conversation.
- *Automate remediation where possible.* Set up automated workflows that not only notify the relevant teams but also initiate predefined remediation processes where

appropriate. This solution is primarily for known high-risk solutions that you can't prevent with automation.

- *Update governance policies and enforcement mechanisms.* Based on the insights gained from the noncompliance event, update your governance policies and enforcement mechanisms. Updates might involve tightening policy definitions, enhancing monitoring capabilities, or refining alert thresholds to improve detection and response times.

Audit cloud governance regularly

Even with automated monitoring, conduct periodic manual reviews and audits to validate compliance monitoring processes and ensure that automation tools are functioning correctly. To audit cloud governance, follow these recommendations:

- *Conduct internal audits.* Conduct regular internal audits to assess compliance with governance policies.
- *Conduct external audits.* Engage external auditors as required to validate compliance with legal and regulatory requirements. Ensure that you consult with legal experts to confirm that your governance policies are in accordance with the applicable laws and regulations in your region.

Next steps

Cloud governance is an ongoing process that requires continuous attention. Consistently repeat the governance process of assessing risks, documenting governance policies, enforcing those policies, and monitoring the effectiveness of the enforcement. The cloud governance team should also work through the cloud governance process whenever they identify new cloud risks.

[Cloud governance overview](#)

[Cloud Adoption Framework overview](#)

Feedback

Was this page helpful?

 Yes

 No

Cloud management in the Cloud Adoption Framework

Article • 12/01/2022

Delivering on a [cloud strategy](#) requires solid planning, readiness, and adoption. But it's the ongoing operation of the digital assets that delivers tangible business outcomes. Without a plan for reliable, well-managed operations of the cloud solutions, those efforts will yield little value. The following exercises help develop the business and technical approaches needed to provide cloud management that powers ongoing operations.

Get started

To prepare you for this phase of the cloud adoption lifecycle, the framework suggests the following exercises:

- 1 **Define business commitments:** Document supported workloads to establish operational commitments with the business and agree on cloud management investments for each workload.
- 2 **Establish a management baseline:** Define the criticality classifications, cloud management tools, and processes required to deliver your minimum commitment to operations management.
- 3 **Expand the management baseline:** Based on business commitments and operations decisions, make use of the included best practices to implement the required cloud management tooling.
- 4 **Advanced operations and design principles:** Platforms or workloads that require a higher level of business commitment might require a deeper architecture review to deliver on resiliency and reliability commitments.

The preceding steps create actionable approaches to deliver on the Manage methodology of the Cloud Adoption Framework.

Manage

Business alignment

Criticality



Document the criticality and relative business value of each workload.

Impact



Establish clear performance expectations and business interruption time/value metrics.

Commitment



Document, track, and report on commitments to cost and performance

Cloud Operations Disciplines



Inventory and visibility

Establish a defined inventory of assets. Develop visibility into the asset telemetry.



Operational compliance

Manage configuration drift and standards. Apply management automation and controls.



Protect and recover

Implement solutions to minimize performance interruptions and ensure rapid recovery, when needed.



Platform operations

Customize operations to improve performance of the common platforms that support multiple workloads.



Workload operations

Understand workload telemetry and align workload operations to performance and reliability commitments.

As discussed in the [business alignment](#) article, not all workloads are mission critical.

Within any portfolio are various degrees of operational management needs. Business alignment efforts aid in capturing the business impact and negotiating management costs with the business, to ensure the most appropriate operational management processes and tools.

The guidance in the manage section of the Cloud Adoption Framework serves two purposes:

- Provides examples of actionable operations management approaches that represent common experiences often encountered by customers.
- Helps you create personalized management solutions based on business commitments.

This content is intended for use by the cloud operations team. It's also relevant to cloud architects who need to develop a strong foundation in cloud operations or cloud design principles.

The content in the Cloud Adoption Framework affects the business, technology, and culture of enterprises. This section of the Cloud Adoption Framework interacts heavily with IT operations, IT governance, finance, line-of-business leaders, networking, identity, and cloud adoption teams. Various dependencies on these personnel require a facilitative approach by the cloud architects who are using this guidance. Facilitation with these teams is seldom a one-time effort.

The cloud architect serves as the thought leader and facilitator to bring these audiences together. The content in this collection of guides is designed to help the cloud architect facilitate the right conversation, with the right audience, to drive necessary decisions. Business transformation that's empowered by the cloud depends on the cloud architect to help guide decisions throughout the business and IT.

Each section of the Cloud Adoption Framework represents a different specialization or variant of the cloud architect role. This section of the Cloud Adoption Framework is designed for cloud architects with a passion for operations and management of deployment solutions. Within this framework, these specialists are referred to frequently as *cloud operations*, or collectively as the *cloud operations team*.

If you want to follow this guide from beginning to end, this content aids in developing a robust cloud operations strategy. The guidance walks you through the theory and implementation of such a strategy.

You can also apply the methodology to [establish clear business commitments](#).

Azure Management Guide: Before you start

Article • 07/03/2023

There are many elements of management that need to be addressed before you define your management strategy. First, it's important to align with your strategy. As you began your cloud journey, you defined some critical outcomes for your business. These outcomes ranged from fiscal goals to operations, sustainability, security, and compliance. As you build your management baseline, ensure that these outcomes are incorporated and addressed. Connecting your management and operations to your strategy and plan will ensure that you have full alignment and accountability across your organization. See the section that covers [business outcomes](#).

The Azure Management Guide helps Azure customers create a management baseline to establish resource consistency across Azure. This guide outlines the basic tools needed for any Azure production environments, especially environments that host sensitive data. For more information, best practices, and considerations related to preparing your cloud environment, see the [readiness section](#) of the Cloud Adoption Framework.

Scope of this guide

This guide teaches you how to establish tooling for a management baseline. It also outlines ways to extend the baseline or build resiliency beyond the baseline.

- ✓ **Inventory and visibility:** Create an inventory of assets across multiple clouds.
 - Develop visibility into the run state of each asset.
- ✓ **Operational compliance:** Establish controls and processes to ensure each state is properly configured and running in a well-governed environment.
- ✓ **Protect and recover:** Ensure all managed assets are protected and can be recovered using baseline management tooling.
- ✓ **Enhanced baseline options:** Evaluate common additions to the baseline that might meet business needs.
- ✓ **Platform operations:** Extend the management baseline with a well-defined service catalog and centrally managed platforms.
- ✓ **Workload operations:** Extend the management baseline to include a focus on mission-critical workloads.

Management baseline

A management baseline is the minimum set of tools and processes that should be applied to every asset in an environment. Several additional options can be included in the management baseline. The next few articles accelerate cloud management capabilities by focusing on the minimum options necessary instead of on all of the available options.

The next step is [Inventory and visibility](#).

Inventory and visibility in Azure

Article • 10/14/2024

Inventory and visibility is the first of three disciplines in a cloud management baseline.



Basic level of cloud management for non-critical, production workloads

This discipline comes first because collecting proper operational data is vital when you make decisions about operations. Cloud management teams must understand what is managed and how well those assets are operated. This article describes the different tools that provide both an inventory and visibility into the inventory's run state.

For any enterprise-grade environment, the following table outlines the suggested minimum for a management baseline.

[] Expand table

Process	Tool	Purpose
Monitor health of Azure services	Azure Service Health	Health, performance, and diagnostics for services running in Azure
Monitoring centralization	Azure Monitor	Central monitoring of operational data and trends
Virtual machine monitoring	Azure Monitor Agent	Monitoring data from the guest operating system of Azure and hybrid virtual machines
Virtual machine inventory and change tracking	Change Tracking and Inventory in Azure Automation	Inventory VMs and monitor changes for guest OS level
Subscription monitoring	Azure activity log	Monitoring change at the subscription

Process	Tool	Purpose
		level
Guest OS monitoring	Azure Monitor for VMs	Monitoring changes and performance of VMs
Network monitoring	Azure Network Watcher	Monitoring network changes and performance
DNS monitoring	DNS Analytics	Security, performance, and operations of DNS

Azure Service Health

Azure Service Health provides a personalized view of the health of your Azure services and regions. Information about active issues is posted to Azure Service Health to help you understand the effect on your resources. Regular updates keep you informed as issues are resolved.

We also publish planned maintenance events to Azure Service Health so you know about changes that can affect resource availability. Set up Service Health alerts to notify you when service issues, planned maintenance, or other changes might affect your Azure services and regions.

Azure Service Health includes:

- **Azure status:** A global view of the health of Azure services.
- **Service health:** A personalized view of the health of your Azure services.
- **Resource health:** A deeper view of the health of your individual resources.

To set up Service Health alerts, go to the [Azure portal](#).

Learn more

For more information, see [Azure Service Health](#).

Azure Monitor

Azure Monitor provides a single unified hub for all monitoring and diagnostics data in Azure and gives you visibility across your resources. With Azure Monitor, you can find and fix problems and optimize performance. You can also understand customer behavior.

- **Monitor and visualize metrics.** Metrics are numerical values available from Azure resources. They help you understand the health of your systems. Customize charts for your dashboards, and use workbooks for reporting.
- **Query and analyze logs.** Logs include activity logs and diagnostic logs from Azure. Collect more logs from other monitoring and management solutions for your cloud or on-premises resources. Log Analytics provides a central repository to aggregate all of this data. From there, you can run queries to help troubleshoot issues or to visualize data.
- **Set up alerts and actions.** Alerts notify you of critical conditions. Corrective actions can be taken based on triggers from metrics, logs, or service-health issues. You can set up different notifications and actions and can also send data to your IT service management tools.

Start monitoring your:

- [Applications](#)
- [Containers](#)
- [Virtual machines](#)
- [Networks](#)

To monitor other resources, find other solutions in Azure Marketplace.

To explore Azure Monitor, go to the [Azure portal](#).

Learn more

To learn more, see [Azure Monitor documentation](#).

Azure Monitor Agent

Azure Monitor Agent (AMA) collects monitoring data from the guest operating system of Azure and hybrid virtual machines and delivers it to Azure Monitor for use by features, insights, and other services such as Microsoft Sentinel and Microsoft Defender for Cloud.

Learn more

To learn more, see the [Azure Monitor Agent overview](#).

Onboard solutions

To enable solutions, you need to configure the Log Analytics workspace. Onboarded Azure VMs and on-premises servers get the solutions from the Log Analytics workspaces they're connected to.

There are two approaches to onboarding:

- [Single VM](#)
- [Entire subscription](#)

Each article guides you through a series of steps to onboard these solutions:

- Azure Update Manager
- Change Tracking and Inventory in Azure Automation
- Azure activity log
- Azure Log Analytics Agent Health
- Antimalware assessment
- Azure Monitor for VMs
- Microsoft Defender for Cloud

Each of the previous steps helps establish inventory and visibility.

Microsoft Cloud for Sovereignty transparency logs

If you're using Microsoft Cloud for Sovereignty, you can use [transparency logs](#) to understand when Microsoft engineers access your resources. These logs help with sovereignty compliance and other regulatory requirements.

Feedback

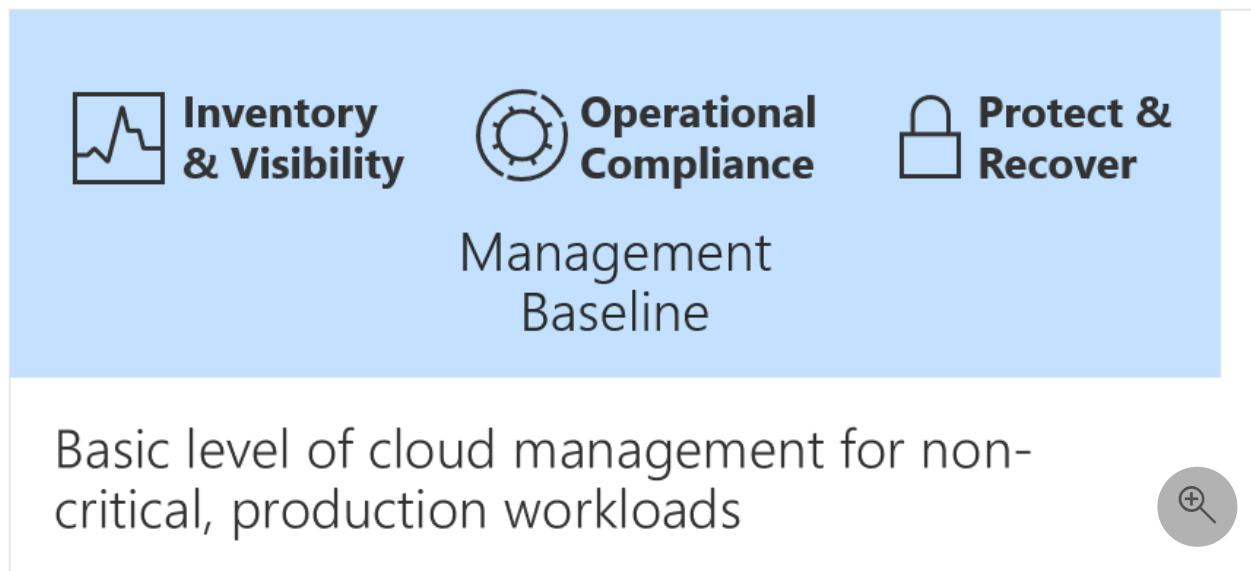
Was this page helpful?

 Yes	 No
---	--

Operational compliance in Azure

Article • 11/15/2024

Operational compliance is the second discipline in any cloud management baseline.



Basic level of cloud management for non-critical, production workloads

Improving operational compliance reduces the likelihood of an outage related to configuration drift or vulnerabilities related to systems being improperly patched.

For any enterprise-grade environment, this table outlines the suggested minimum for a management baseline.

[\[\] Expand table](#)

Process	Tool	Purpose
Patch management	Azure Automation Update Management	Management and scheduling of updates
Policy enforcement	Azure Policy	Automated policy enforcement to ensure environment and guest compliance
Environment configuration	Infrastructure as code (IaC)	Automated environment creation, configuration, and to avoid configuration drift
Resource configuration	Desired State Configuration (DSC)	Automated configuration on guest OS and some aspects of the environment

Update Management

Computers that are managed by the Update Management solution for Azure Automation use the following configurations to do assessment and update

deployments:

- Log Analytics agent for Windows or Linux.
- PowerShell DSC for Linux.
- Azure Automation Hybrid Runbook Worker.
- Microsoft Update or Windows Server Update Services (WSUS) for Windows computers.

For more information, see [Update Management solution for Azure Automation](#).

Warning

Before using Update Management, you must onboard virtual machines or an entire subscription into Log Analytics and Azure Automation.

There are two approaches to onboarding:

- [Single VM](#)
- [Entire subscription](#)

You should follow one before proceeding with Update Management.

Manage updates

To apply a policy to a resource group:

1. Go to [Azure Automation](#).
2. Select **Automation accounts**, and choose one of the listed accounts.
3. Go to **Configuration Management**.
4. Use **State Configuration (DSC)** to control the state and operational compliance of the managed VMs.

Azure Policy

Azure Policy is used throughout governance processes. It's also highly valuable within cloud management processes. Azure Policy can audit and remediate Azure resources and can also audit and configure settings inside a machine. The validation is performed by the machine configuration extension and client. The extension, through the client, validates settings like:

- Operating system configuration.
- Application configuration or presence.

- Environment settings.

An important part of this process is maintaining and updating Azure Policy assignments as your governance process requires. Using IaC can help you update and maintain your policy infrastructure. For more information, see [Use IaC to update Azure landing zones](#).

Apply a policy

To apply a policy to a resource group:

1. Go to [Azure Policy](#).
2. Select **Assign a policy**.

Learn more

To learn more, see:

- [Azure Policy](#)
- [Azure Policy machine configuration](#)
- [Cloud Adoption Framework: Define corporate policy](#)

Feedback

Was this page helpful?



Protect and recover in Azure

Article • 12/01/2022

Protect and recover is the third and final discipline in any cloud-management baseline.



Basic level of cloud management for non-critical, production workloads

In [Operational compliance in Azure](#) the objective is to reduce the likelihood of a business interruption. The current article aims to reduce the duration and impact of outages that can't be prevented.

For any enterprise-grade environment, this table outlines the suggested minimum for any management baseline:

Process	Tool	Purpose
Protect data	Azure Backup	Back up data and virtual machines in the cloud.
Protect the environment	Microsoft Defender for Cloud	Strengthen security and provide advanced threat protection across your hybrid workloads.

Azure Backup

With Azure Backup, you can back up, protect, and recover your data in the Microsoft cloud. Azure Backup replaces your existing on-premises or offsite backup solution with a cloud-based solution. This new solution is reliable, secure, and cost competitive. Azure Backup can also help protect and recover on-premises assets through one consistent solution.

For data present in Azure, Azure Backup offer varied levels of protection. For example, when backing up key cloud infrastructure pieces such as Azure Virtual Machines and

Azure Files, it offers [Azure Virtual Machines backup](#) and [Azure Files backup](#). For more critical components such as databases running in Azure Virtual Machines, it offers dedicated database backup solutions for [SQL Server](#) and [SAP HANA](#) with far lower RPO.

Review the following section to see how easily you can enable backup for Azure Virtual Machines.

Enable backup for an Azure VM

1. In the Azure portal, select **Virtual machines**, then select the VM you want to backup.
2. On the **Operations** pane, select **Backup**.
3. Create or select an existing Azure Recovery Services vault.
4. Select **Create (or edit) a new policy**.
5. Configure the schedule and retention period.
6. Select **OK**.
7. Select **Enable backup**.

For more details about Azure Backup, see [Overview of Azure Backup](#).

Azure Site Recovery

Azure Site Recovery is a critical component in your disaster recovery strategy.

Site Recovery replicates VMs and workloads that are hosted in a primary Azure region. It replicates them to a copy that is hosted in a secondary region. When an outage occurs in your primary region, you fail over to the copy running in the secondary region. You then continue to access your applications and services from there. This proactive approach to recovery can significantly reduce recovery times. When the recovery environment is no longer needed, production traffic can fall back to the original environment.

Replicate an Azure VM to another region with Site Recovery

The following steps outline the process to use Site Recovery for Azure-to-Azure replication, which is replication of an Azure VM to another region.

 Tip

Depending on your scenario, the exact steps might differ slightly.

Enable replication for the Azure VM

1. In the Azure portal, select **Virtual machines**, then select the VM you want to replicate.
2. On the **Operations** pane, select **Disaster recovery**.
3. Select **Configure disaster recovery > Target region**, and choose the target region to which you'll replicate.
4. For this quickstart, accept the default values for all other options.
5. Select **Enable replication**, which starts a job to enable replication for the VM.

Verify settings

After the replication job has finished, you can check the replication status, verify replication health, and test the deployment.

1. In the VM menu, select **Disaster recovery**.
2. Verify replication health, the recovery points that have been created, and source and target regions on the map.

Learn more

- [Azure Site Recovery overview](#)
- [Replicate an Azure VM to another region](#)

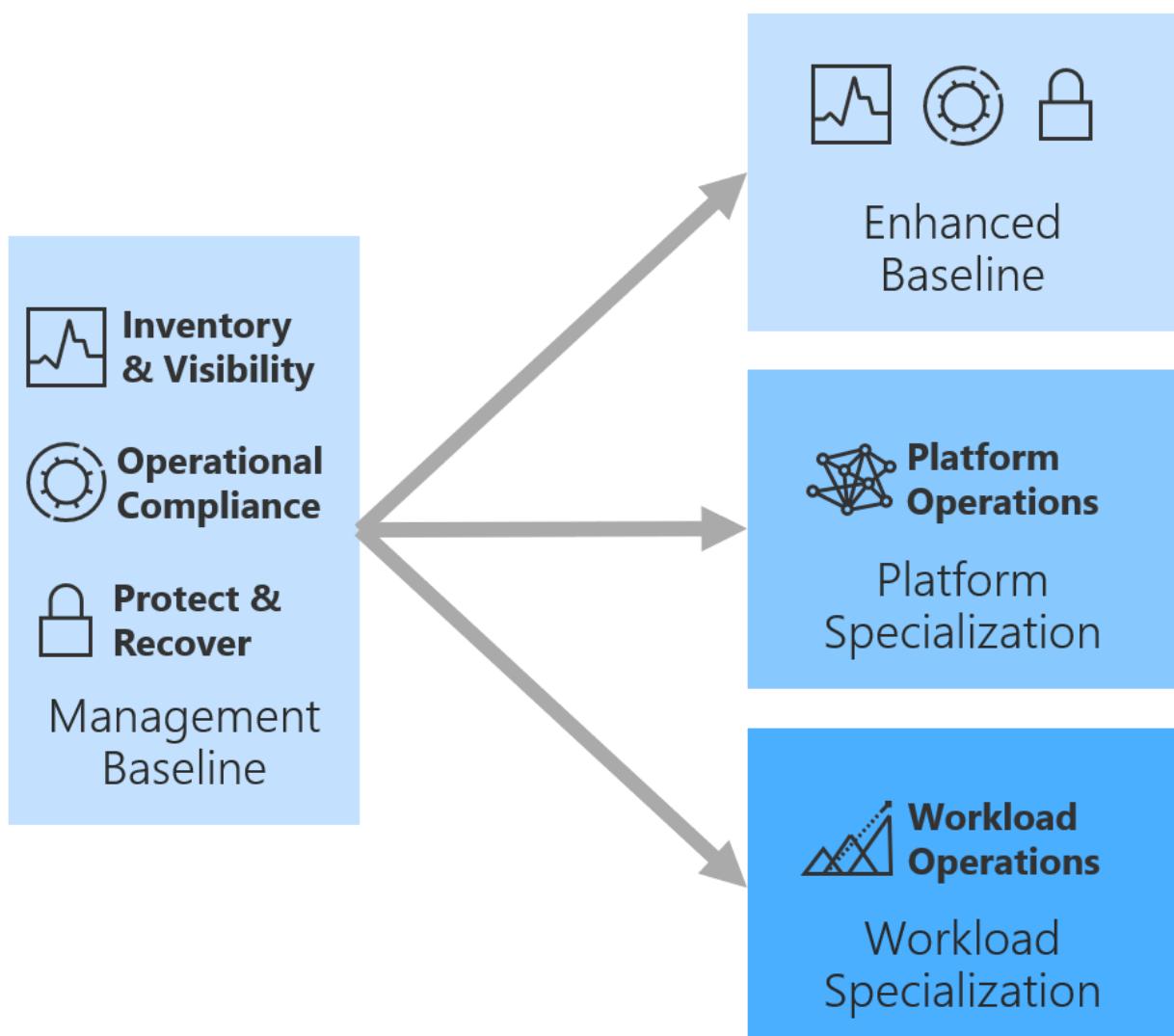
Enhanced management baseline in Azure

Article • 07/18/2024

The first three cloud management disciplines describe a management baseline. The preceding articles in this guide outline a minimum viable product (MVP) for cloud management services, which is referred to as a management baseline. This article outlines a few common improvements to the baseline.

The purpose of a management baseline is to create a consistent offering that provides a minimum level of business commitment for **all** supported workloads. With this baseline of common, repeatable management offerings, the team can deliver highly optimized operational management with minimal deviation.

However, you might need a greater commitment to the business beyond the standard offering. The following image and list show three ways to go beyond the management baseline.



- **Enhanced management baseline:**
 - Add enhancements to the management baseline, when most workloads in the portfolio have a shared requirement.
 - Improved business commitments using additional cloud-native operations tools and processes.
 - Baseline enhancements should have no impact on the architecture of specific workloads.
- **Workload operations:**
 - Largest per-workload operations investment.
 - Highest degree of resiliency.
 - Suggested for the approximately 20 percent of workloads that drive business value.
 - Typically reserved for high-criticality or mission-critical workloads.
- **Platform operations:**
 - Operations investment is spread across many workloads.
 - Resiliency improvements affect all workloads that use the defined platform.
 - Suggested for the approximately 20 percent of platforms that have highest criticality.
 - Typically reserved for medium-criticality to high-criticality workloads.

Both workload operations and platform operations require changes to design and architecture principles. Those changes can take time and might result in increased operating expenses. To reduce the number of workloads that require such investments, an enhanced management baseline can provide enough of an improvement to the business commitment.

This table outlines a few processes, tools, and potential effects common in customers' enhanced management baselines:

[] Expand table

Discipline	Process	Tool	Potential impact	Learn more
Inventory and visibility	Service change tracking	Azure Resource Graph	Greater visibility into changes to Azure services might help detect negative effects sooner or remediate faster.	Overview of Azure Resource Graph
Inventory and visibility	Visualize Data	Microsoft Sentinel	Instant visualization and analysis of data	Sentinel visualize collected data

Discipline	Process	Tool	Potential impact	Learn more
Inventory and visibility	IT Service Management (ITSM) integration	IT Service Management Connector	Automated ITSM connection creates awareness sooner.	IT Service Management Connector (ITSMC)
Operational compliance	Operations automation	Azure Automation	Automate operational compliance for faster and more accurate response to change.	See the following sections
Operational compliance	Zero trust	Microsoft Sentinel	Zero Trust workbook uses the full breadth of Microsoft security offerings	Sentinel Zero trust Workbook
Operational compliance	Performance automation	Azure Automation	Automate operational compliance with performance expectations to resolve common resource specific scaling or sizing issues.	See the following sections
Operational compliance	Multicloud operations	Azure Automation Hybrid Runbook Worker	Automate operations across multiple clouds.	Hybrid Runbook Worker overview
Operational compliance	Guest automation	Desired State Configuration (DSC)	Code-based configuration of guest operating systems to reduce errors and configuration drift.	DSC overview
Protect and recover	Breach notification	Microsoft Defender for Cloud	Extend protection to include security-breach recovery triggers.	See the following sections
Protect and recover	Threat Hunting	Microsoft Sentinel	Built in hunting queries that help you detect and protect against malicious activity	Sentinel Threat Hunting

Azure Automation

[Azure Automation](#) provides a centralized system for the management of automated controls. In Azure Automation, you can run simple remediation, scale, and optimization

processes in response to environmental metrics. These processes reduce the overhead associated with manual incident processing.

Most importantly, automated remediation can be delivered in near-real-time, significantly reducing interruptions to business processes. A study of the most common business interruptions identifies activities within your environment that could be automated.

Runbooks

The basic unit of code for delivering automated remediation is a runbook. Runbooks contain the instructions for remediating or recovering from an incident.

To create or manage runbooks:

1. Sign in to the Azure [portal](#) ↗
2. Go to **Azure Automation**.
3. Select **Automation accounts** and choose one of the listed accounts.
4. Go to **Process automation**, select **Runbooks** to open the list of runbooks.
5. With the options presented, you can create or manage runbooks, schedules, and other automated remediation functionality.

Microsoft Defender for Cloud

Microsoft Defender for Cloud also plays an important part in your protect-and-recover strategy. It can help you monitor the security of your machines, networks, storage, data services, and applications.

Microsoft Defender for Cloud provides advanced threat detection by using machine learning and behavioral analytics to help identify active threats targeting your Azure resources. It also provides threat protection that blocks malware and other unwanted code, and it reduces the surface area exposed to brute force and other network attacks.

When Microsoft Defender for Cloud identifies a threat, it triggers a security alert with steps you need for responding to an attack. It also provides a report with information about the detected threat.

Microsoft Defender for Cloud is offered in two tiers: Free and Standard. Features like security recommendations are available in the Free tier. The Standard tier provides additional protection like advanced threat detection and protection across hybrid cloud workloads.

To explore Microsoft Defender for Cloud, go to the [Azure portal](#) ↗.

Learn more

To learn more, see [Microsoft Defender for Cloud documentation](#).

Microsoft Sentinel

Microsoft Sentinel is a cloud-native security information event management (SIEM) and security orchestration automated response (SOAR) solution that plays a role, not only in your Enhanced management baseline in Azure but also in the Enhanced Baseline, Platform Operations and Workload Operations.

Microsoft Sentinel allows you to Collect Data, Detect Threats, Investigate Incidents, and Respond using Automation. Upon enabling the solution the ability to connect and collect data from Azure, On-premises, or any other cloud provider becomes available. There are over a hundred Data Connectors available including Office 365 Audit Logs, Azure Activity Logs, Cisco Umbrella, Trend Micro TippingPoint, Sophos Cloud Optix, VMware ESXi, and many others that simplify your integration into Sentinel from existing investments.

Microsoft Sentinel can be enabled at no additional cost on an Azure Monitor Log Analytics workspace for the first 31-days. See more [Microsoft Sentinel Pricing](#).

Learn more about Microsoft Sentinel

To explore Microsoft Sentinel, go to the [Azure portal](#).

To learn more, see [Microsoft Sentinel documentation](#).

Want to become a Microsoft Sentinel Ninja, see [Microsoft Sentinel Ninja Training](#).

Feedback

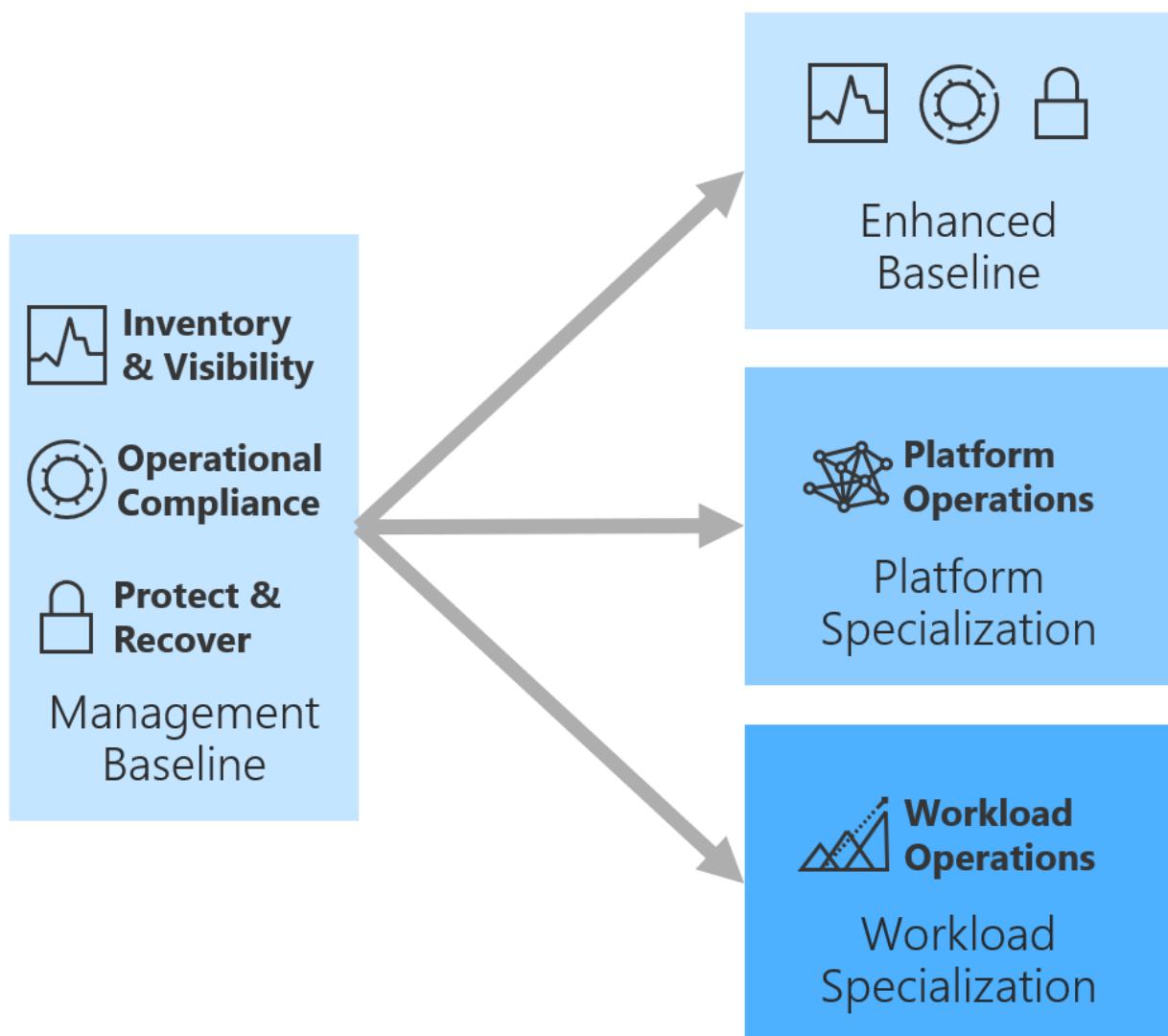
Was this page helpful?



Platform specialization for cloud management

Article • 02/16/2023

Much like the enhanced management baseline, platform specialization is extension beyond the standard management baseline. See the following image and list that show the ways to expand the management baseline. This article addresses the platform specialization options.



- **Workload operations:** The largest per-workload operations investment and the highest degree of resiliency. We suggest workload operations for the approximately 20% of workloads that drive business value. This specialization is usually reserved for high criticality or mission-critical workloads.
- **Platform operations:** Operations investment is spread across many workloads. Resiliency improvements affect all workloads that use the defined platform. We suggest platform operations for the approximately 20% of platforms that have the

highest criticality. This specialization is usually reserved for medium to high criticality workloads.

- **Enhanced management baseline:** The relatively lowest operations investment. This specialization slightly improves business commitments by using additional cloud-native operations tools and processes.

Both workload and platform operations require changes to design and architecture principles. Those changes can take time and might result in increased operating expenses. To reduce the number of workloads requiring such investments, an enhanced management baseline might provide enough of an improvement to the business commitment.

This table outlines a few common processes, tools, and potential effects common in customers' enhanced management baselines:

Process	Tool	Purpose	Suggested management level
Improve system design	Microsoft Azure Well-Architected Framework	Improving the architectural design of the platform to improve operations	N/A
Automate remediation	Azure Automation	Responding to advanced platform data with platform-specific automation	Platform operations
Service catalog	Managed applications center	Providing a self-service catalog of approved solutions that meet organizational standards	Platform operations
Container performance	Azure Monitor for containers	Monitoring and diagnostics of containers	Platform operations
Platform as a service (PaaS) data performance	Azure SQL Analytics	Monitoring and diagnostics for PaaS databases	Platform operations
Infrastructure as a service (IaaS) data performance	SQL Server Health Check	Monitoring and diagnostics for IaaS databases	Platform operations

High-level process

Platform specialization consists of a disciplined execution of the following four processes in an iterative approach. Each process is explained in more detail in later

sections of this article.

- **Improve system design:** Improve the design of common systems or platforms to effectively minimize interruptions.
- **Automate remediation:** Some improvements aren't cost effective. In such cases, it might make more sense to automate remediation and reduce the effect of interruptions.
- **Scale the solution:** As systems design and automated remediation are improved, those changes can be scaled across the environment through the service catalog.
- **Continuous improvement:** Different monitoring tools can be used to discover incremental improvements. These improvements can be addressed in the next pass of system design, automation, and scale.

Improve system design

Improving system design is the most effective approach to improving operations of any common platform. Through system-design improvements, stability can increase and business interruptions can decrease. Design of individual systems is beyond the scope of the environment view that's taken throughout the Cloud Adoption Framework.

As a complement to this framework, the [Microsoft Azure Well-Architected Framework](#) provides guiding tenets for improving the quality of a platform or a specific workload. The framework focuses on improvement across five pillars of architecture excellence:

- **Cost optimization:** Manage costs to maximize the value delivered.
- **Operational excellence:** Follow operational processes that keep a system running in production.
- **Performance efficiency:** Scale systems to adapt to changes in load.
- **Reliability:** Design systems to recover from failures and continue to function.
- **Security:** Protect applications and data from threats.

Technical debt and architectural flaws cause most business interruptions. For existing deployments, you can view system-design improvements as payments against existing technical debt. For new deployments, you can view those improvements as avoidance of technical debt.

The following **Automated remediation** tab shows ways to remediate technical debt that can't or shouldn't be addressed.

Learn more about the [Microsoft Azure Well-Architected Framework](#) to improve system design.

As system design improves, return to this article to find new opportunities to improve and scale those improvements across your environment.

Automated remediation

Some technical debt can't be addressed. Resolution might be too expensive to correct or might be planned but have a long project duration. The business interruption might not have a significant business effect. Or the business priority might be to recover quickly instead of investing in resiliency.

When resolution of technical debt isn't the desired approach, automated remediation is commonly the next step. Using Azure Automation and Azure Monitor to detect trends and provide automated remediation is the most common approach to automated remediation.

For guidance on automated remediation, see [Azure Automation and alerts](#).

Scale the solution with a service catalog

A well-managed service catalog is the cornerstone of platform specialization and platform operations. Use of a catalog is how improvements to systems design and remediation are scaled across an environment.

The cloud platform team and cloud automation team align to create repeatable solutions to the most common platforms in any environment. But if those solutions aren't consistently used, cloud management can provide little more than a baseline offering.

To maximize adoption and minimize maintenance overhead of any optimized platform, you should add the platform to an Azure service catalog. You can deploy each application in the catalog for internal consumption via the service catalog or as a marketplace offering for external consumers.

For instructions on publishing to a service catalog, see the article series on [publishing to a service catalog](#).

Deploy applications from the service catalog

1. In the Azure portal, go to [Managed applications center \(preview\)](#).
2. On the **Browse** pane, select **Service Catalog applications**.
3. Select **+ Add** to choose an application definition from your company's service catalog.

Any managed applications you're servicing are displayed.

Manage service catalog applications

1. In the Azure portal, go to [Managed applications center \(preview\)](#).
2. On the **Service** pane, select **Service Catalog applications**.

Any managed applications you're servicing are displayed.

Continuous improvement

Platform specialization and platform operations both depend on strong feedback loops among adoption, platform, automation, and management teams. Grounding those feedback loops in data helps each team make wise decisions. For platform operations to achieve long-term business commitments, it's important to use insights specific to the centralized platform.

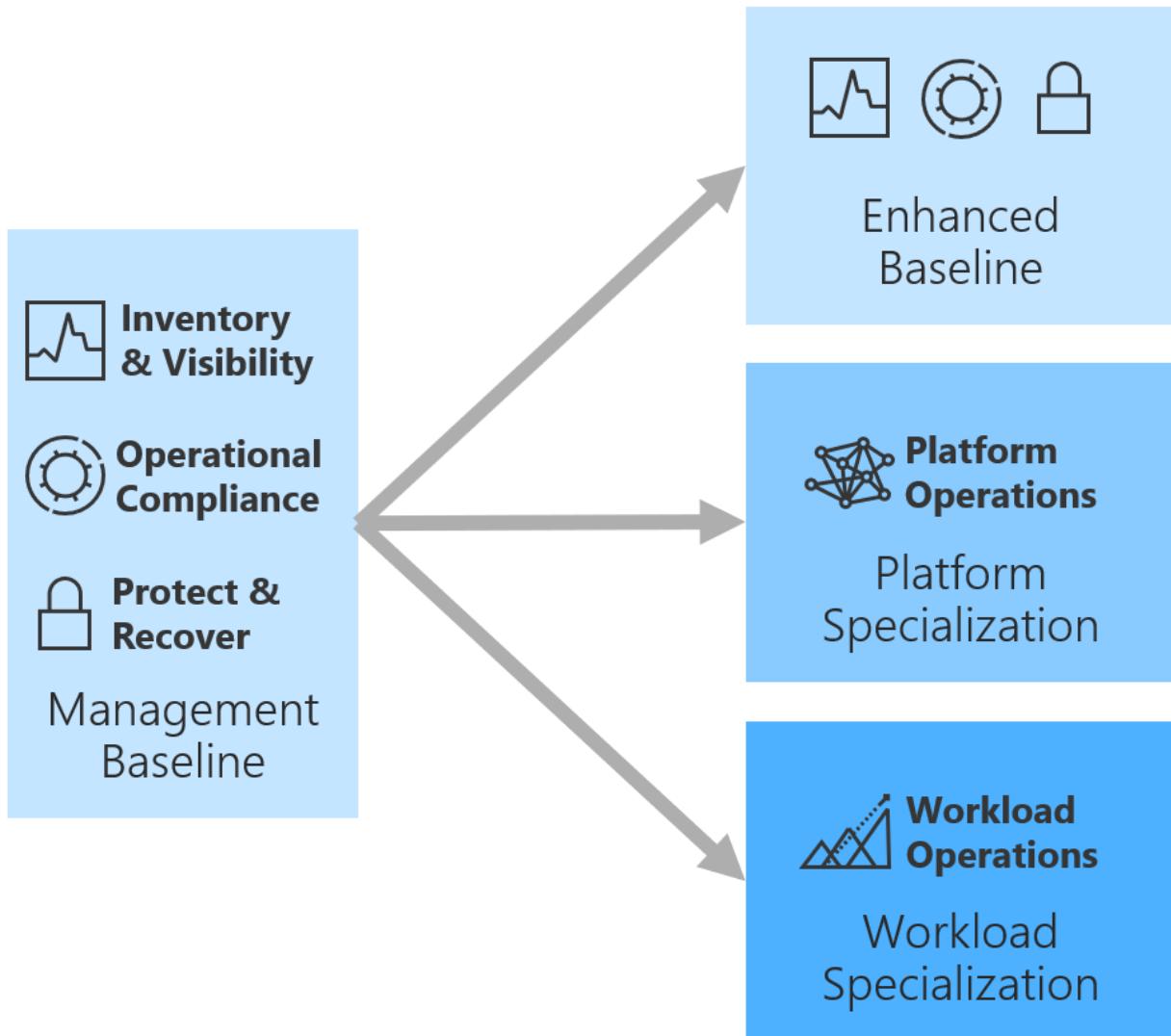
Containers and SQL Server are the two most common centrally managed platforms. These articles can help you get started with continuous-improvement data collection on those platforms:

- [Container performance](#)
- [PaaS database performance](#)
- [IaaS database performance](#)

Workload specialization for cloud management

Article • 02/16/2023

Workload specialization builds on the concepts outlined in [Platform specialization](#).



- **Workload operations:** The largest per-workload operations investment and highest degree of resiliency. We suggest workload operations for the approximately 20% of workloads that drive business value. This specialization is usually reserved for high criticality or mission-critical workloads.
- **Platform operations:** Operations investment is spread across many workloads. Resiliency improvements affect all workloads that use the defined platform. We suggest platform operations for the approximately 20% of platforms that have the highest criticality. This specialization is usually reserved for medium to high criticality workloads.

- **Enhanced management baseline:** The relatively lowest operations investment. This specialization slightly improves business commitments by using additional cloud-native operations tools and processes.

High-level process

Workload specialization consists of a disciplined execution of the following four processes in an iterative approach. Each process is explained in more detail in [Platform specialization](#).

- **Improve system design:** Improve the design of a specific workload to effectively minimize interruptions.
- **Automate remediation:** Some improvements aren't cost effective. In such cases, it might make more sense to automate remediation and reduce the effect of interruptions.
- **Scale the solution:** As you improve systems design and automated remediation, you can scale those changes across the environment through the service catalog.
- **Continuous improvement:** You can use different monitoring tools to discover incremental improvements. These improvements can be addressed in the next pass of system design, automation, and scale.

Cultural change

Workload specialization often triggers a cultural change in traditional IT build processes that focus on delivering a management baseline, enhanced baselines, and platform operations. Those types of offerings can be scaled across the environment. Workload specialization is similar in execution to platform specialization. But unlike common platforms, the specialization required by individual workloads often doesn't scale.

When workload specialization is required, operational management commonly evolves beyond a centralized IT perspective. The approach suggested in Cloud Adoption Framework is a distribution of cloud management functionality.

In this model, operational tasks like monitoring, deployment, DevOps, and other innovation-focused functions shift to an application-development or business-unit organization. The cloud platform team and the core cloud monitoring team still delivers on the management baseline across the environment.

Those centralized teams also guide and instruct workload-specialized teams on operations of their workloads. But the day-to-day operational responsibility falls on a

cloud management team that is managed outside of IT. This type of distributed control is one of the primary indicators of maturity in a cloud center of excellence.

Beyond platform specialization: Application Insights

Greater detail on the specific workload is required to provide clear workload operations. During the continuous improvement phase, Application Insights will be a necessary addition to the cloud management toolchain.

Requirement	Tool	Purpose
Application monitoring	Application Insights	Monitoring and diagnostics for applications
Performance, availability, and usage	Application Insights	Advanced application monitoring with the application dashboard, composite maps, usage, and tracing

Deploy Application Insights

1. In the Azure portal, go to [Application Insights](#).
2. Select **+ Add** to create an Application Insights resource to monitor your live web application.
3. Follow the on-screen prompts.

See the [Azure Monitor Application Insights hub](#) for guidance on configuring your application for monitoring.

Monitor performance, availability, and usage

1. In the Azure portal, search for [Application Insights](#).
2. Choose one of the Application Insights resources from the list.

Application Insights contains different kinds of options for monitoring performance, availability, usage, and dependencies. These views of the application data add clarity to the continuous-improvement feedback loop.

Establish operational management practices in the cloud

Article • 12/01/2022

Cloud adoption is a catalyst for enabling business value. However, real business value is realized through ongoing, stable operations of the technology assets deployed to the cloud. This section of the Cloud Adoption Framework guides you through various transitions into operational management in the cloud.

Actionable best practices

Modern operations management solutions create a multicloud view of operations. Assets managed through the following best practices may live in the cloud, in an existing datacenter, or even in a competing cloud provider. Currently, the framework includes two best-practices references to guide operations management maturity in the cloud:

- [Azure server management](#): An onboarding guide to incorporating the cloud-native tools and services needed to manage operations.
- [Hybrid monitoring](#): Many customers have already made a substantial investment in System Center Operations Manager. For those customers, this guide to hybrid monitoring can help them compare and contrast the cloud-native reporting tools with Operations Manager tooling. This comparison makes it easier to decide which tools to use for operational management.

Cloud operations

Both of these best practices build toward a future-state methodology for operations management, as illustrated in the following diagram:

Manage

Business alignment

Criticality



Document the criticality and relative business value of each workload.

Impact



Establish clear performance expectations and business interruption time/value metrics.

Commitment



Document, track, and report on commitments to cost and performance

Cloud Operations Disciplines



Inventory and visibility

Establish a defined inventory of assets. Develop visibility into the asset telemetry.



Operational compliance

Manage configuration drift and standards. Apply management automation and controls.



Protect and recover

Implement solutions to minimize performance interruptions and ensure rapid recovery, when needed.



Platform operations

Customize operations to improve performance of the common platforms that support multiple workloads.



Workload operations

Understand workload telemetry and align workload operations to performance and reliability commitments.

Business alignment: In the Manage methodology, all workloads are classified by criticality and business value. That classification can then be measured through an impact analysis, which calculates the lost value associated with performance degradation or business interruptions. Using that tangible revenue impact, cloud operations teams can work with the business to establish a commitment that balances cost and performance.

Cloud operations disciplines: After the business is aligned, it's much easier to track and report on the proper disciplines of cloud operations for each workload. Making decisions along each discipline can then be converted to commitment terms that can be easily understood by the business. This collaborative approach makes the business stakeholder a partner in finding the right balance between cost and performance.

- **Inventory and visibility:** At a minimum, operations management requires a means of inventorying assets and creating visibility into the run state of each asset.
- **Operational compliance:** Regular management of configuration, sizing, cost, and performance of assets is key to maintaining performance expectations.
- **Protect and recover:** Minimizing operational interruptions and expediting recovery help the business avoid performance losses and adverse revenue impacts. Detection and recovery are essential aspects of this discipline.
- **Platform operations:** All IT environments contain a set of commonly used platforms. Those platforms could include data stores such as SQL Server or Azure HDInsight. Other common platforms could include container solutions such as Azure Kubernetes Service (AKS). Regardless of the platform, platform operations maturity focuses on customizing operations based on how the common platforms are deployed, configured, and used by workloads.
- **Workload operations:** At the highest level of operational maturity, cloud operations teams can tune operations for critical workloads. For those workloads, available data can assist in automating the remediation, sizing, or protection of workloads based on their utilization.

Additional guidance, such as the [Microsoft Azure Well-Architected Framework](#), can help you make detailed architectural decisions about each workload, within the previously described disciplines.

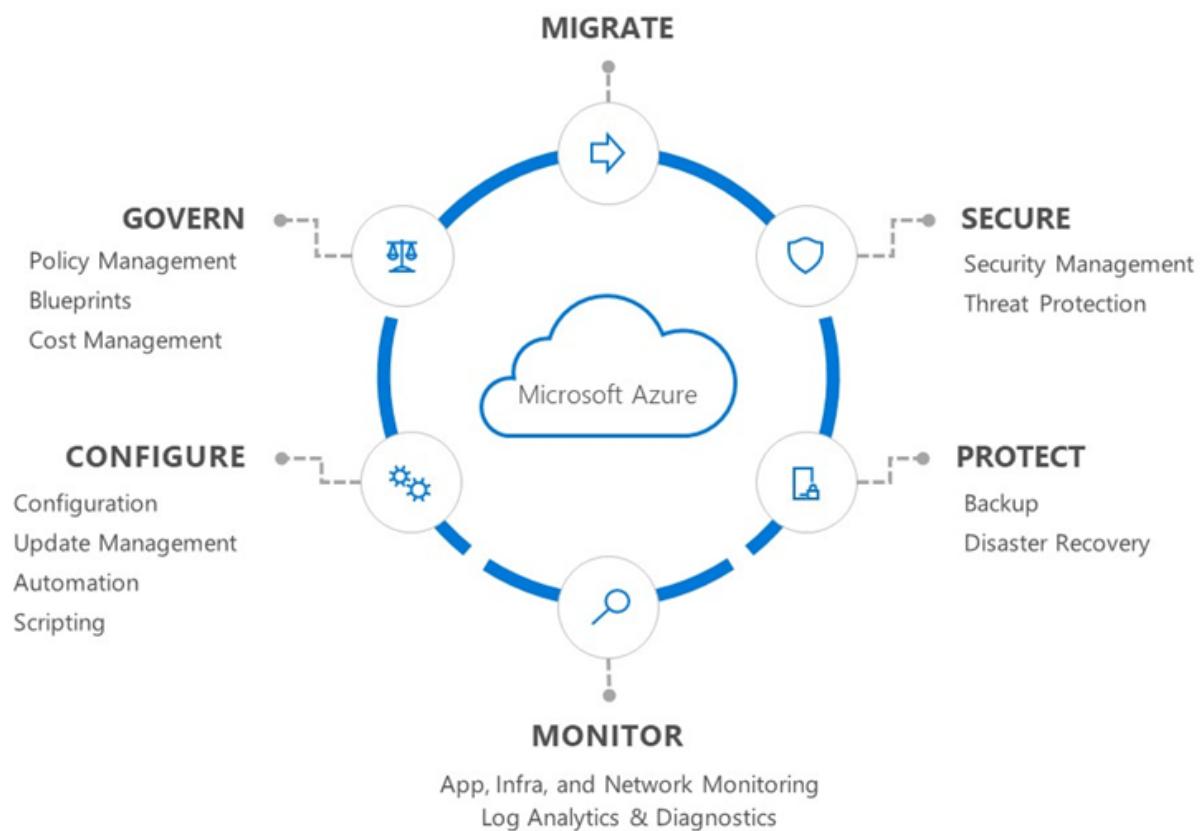
This section of the Cloud Adoption Framework will build on each of the preceding topics to help promote mature cloud operations within your organization.

Overview of Azure server management services

Article • 12/06/2022

Azure server management services provide a consistent experience for managing servers at scale. These services cover both Linux and Windows operating systems. They can be used in production, development, and test environments. The server management services can support Azure IaaS virtual machines, physical servers, and virtual machines that are hosted on-premises or in other hosting environments.

The Azure server management services suite includes the services in the following diagram:



This section of the Microsoft Cloud Adoption Framework provides an actionable and prescriptive plan for deploying server management services in your environment. This plan helps orient you quickly to these services, guiding you through an incremental set of management stages for all environment sizes.

For simplicity, we've categorized this guidance into three stages:



Why use Azure server management services?

Azure server management services offer the following benefits:

- **Native to Azure:** Server management services are built into and natively integrated with Azure Resource Manager. These services are continuously improved to provide new features and capabilities.
- **Windows and Linux:** Windows and Linux machines get the same consistent management experience.
- **Hybrid:** The server management services cover Azure IaaS virtual machines as well as physical and virtual servers that are hosted on-premises or in other hosting environments.
- **Security:** Microsoft devotes substantial resources to all forms of security. This investment not only protects the Azure infrastructure but also extends the resulting technologies and expertise to protect customers' resources wherever they reside.
- **Windows Server and Azure:** Windows Server together with Microsoft Azure cloud management capabilities allows you to get even more out of your Windows Server investments. You can learn more on [Better together Windows Server and Azure Management](#).

Next steps

Familiarize yourself with the [tools, services, and planning](#) involved with adopting the Azure server management suite.

[Prerequisite tools and planning](#)

Phase 1: Prerequisite planning for Azure server management services

Article • 10/24/2024

In this phase, you'll become familiar with the Azure server management suite of services, and plan how to deploy the resources needed to implement these management solutions.

Understand the tools and services

Review [Azure server management tools and services](#) for a detailed overview of:

- The management areas that are involved in ongoing Azure operations.
- The Azure services and tools that help support you in these areas.

You'll use several of these services together to meet your management requirements. These tools are referenced often throughout this guidance.

The following sections discuss the planning and preparation required to use these tools and services.

Log Analytics workspace and Automation account planning

Many of the services you'll use to onboard Azure management services require a Log Analytics workspace and a linked Azure Automation account.

A [Log Analytics workspace](#) is a unique environment for storing Azure Monitor log data. Each workspace has its own data repository and configuration. Data sources and solutions are configured to store their data in particular workspaces. Azure monitoring solutions require all servers to be connected to a workspace, so that their log data can be stored and accessed.

Some of the management services require an [Azure Automation](#) account. You use this account, and the capabilities of Azure Automation, to integrate Azure services and other public systems to deploy, configure, and manage your server management processes.

The following Azure server management services require a linked Log Analytics workspace and Automation account:

- [Update Management](#)
- [Change Tracking and Inventory](#)
- [Hybrid Runbook Worker](#)
- [Desired State Configuration](#)

The second phase of this guidance focuses on deploying services and automation scripts. It shows you how to create a Log Analytics workspace and an Automation account. This guidance also shows you how to use Azure Policy to ensure that new virtual machines are connected to the correct workspace.

The examples in this guidance assume a deployment that doesn't already have servers deployed to the cloud. To learn more about the principles and considerations involved in planning your workspaces, see [Manage log data and workspaces in Azure Monitor](#).

Planning considerations

When preparing the workspaces and accounts that you need for onboarding management services, consider the following issues:

- **Azure geographies and regulatory compliance:** Azure regions are organized into *geographies*. An [Azure geography](#) ensures that data residency, sovereignty, compliance, and resiliency requirements are honored within geographical boundaries. If your workloads are subject to data-sovereignty or other compliance requirements, workspace and Automation accounts must be deployed to regions within the same Azure geography as the workload resources they support.
- **Number of workspaces:** As a guiding principle, create the minimum number of workspaces required per Azure geography. We recommend at least one workspace for each Azure geography where your compute or storage resources are located. This initial alignment helps avoid future regulatory issues when you migrate data to different geographies.
- **Data retention and capping:** You may also need to take data retention policies or data capping requirements into consideration when creating workspaces or Automation accounts. For more information about these principles, and for additional considerations when planning your workspaces, see [Manage log data and workspaces in Azure Monitor](#).
- **Region mapping:** Linking a Log Analytics workspace and an Azure Automation account is supported only between certain Azure regions. For example, if the Log Analytics workspace is hosted in the `East US` region, the linked Automation account must be created in the `East US 2` region to be used with management services. If you have an Automation account that was created in another region, it can't link to a workspace in `East US`. The choice of deployment region can

significantly affect Azure geography requirements. Consult the [region mapping table](#) to decide which region should host your workspaces and Automation accounts.

- **Workspace multihoming:** The Azure Log Analytics agent supports multihoming in some scenarios, but the agent faces several limitations and challenges when running in this configuration. Unless Microsoft has recommended it for your specific scenario, don't configure multihoming on the Log Analytics agent.

Resource placement examples

There are several different models for choosing the subscription in which you place the Log Analytics workspace and Automation account. In short, place the workspace and Automation accounts in a subscription owned by the team that's responsible for implementing the Update Management solution and the Change Tracking and Inventory service.

The following are examples of some ways to deploy workspaces and Automation accounts.

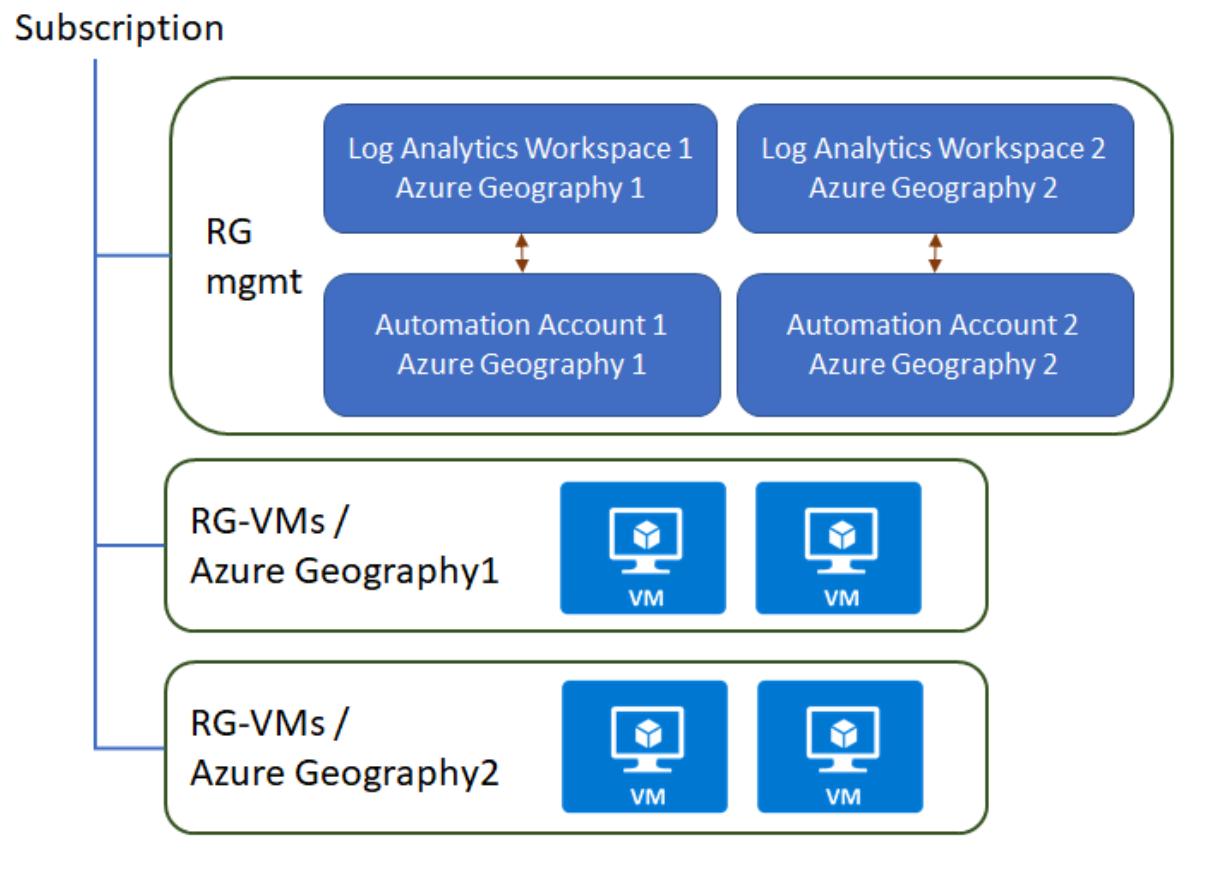
Placement by geography

Small and midsize environments have a single subscription and several hundred resources that span multiple Azure geographies. For these environments, create one Log Analytics workspace and one Azure Automation account in each geography.

You can create a workspace and an Azure Automation account, as one pair, in each resource group. Then, deploy the pair in the corresponding geography to the virtual machines.

Alternatively, if your data-compliance policies don't dictate that resources reside in specific regions, you can create one pair to manage all the virtual machines. We also recommend that you place the workspace and Automation account pairs in separate resource groups to provide more granular Azure role-based access control (Azure RBAC).

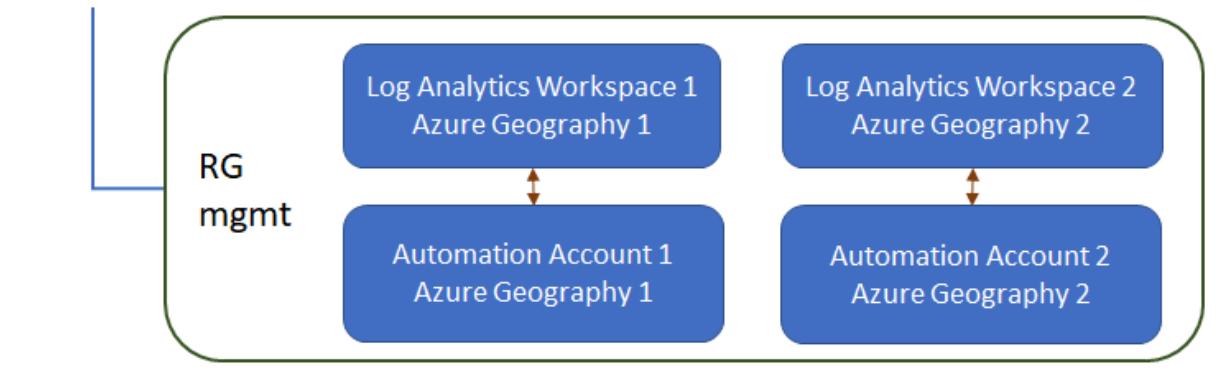
The example in the following diagram has one subscription with two resource groups, each located in a different geography:



Placement in a management subscription

Larger environments span multiple subscriptions and have a central IT team that owns monitoring and compliance. For these environments, create pairs of workspaces and Automation accounts in an IT management subscription. In this model, virtual-machine resources in a geography store their data in the corresponding geography workspace in the IT management subscription. If application teams need to run automation tasks but don't require linked workspace and Automation accounts, they can create separate Automation accounts in their own application subscriptions.

Subscription (Management)



Subscription (Application team)



Subscription (Application team)



Decentralized placement

In an alternative model for large environments, the application development team can be responsible for patching and management. In this case, place the workspace and Automation account pairs in the application team subscriptions alongside their other resources.

Subscription (Application team)

RG - VMs /
Azure Geography 1

Log Analytics Workspace 1
Azure Geography 1



Automation Account 1
Azure Geography 1



Subscription (Application team)

RG - VMs /
Azure Geography 2

Log Analytics Workspace 2
Azure Geography 2



Automation Account 2
Azure Geography 2



Create a workspace and Automation account

After you've chosen the best way to place and organize workspace and account pairs, make sure that you've created these resources before starting the onboarding process. The automation examples later in this guidance create a workspace and Automation account pair for you. However, if you want to onboard by using the Azure portal and you don't have an existing workspace and Automation account pair, you'll need to create one.

To create a Log Analytics workspace by using the Azure portal, see [Create a workspace](#). Next, create a matching Automation account for each workspace by following the steps in [Create an Azure Automation account](#).

Next steps

Learn how to [onboard your servers](#) to Azure server management services.

[Onboard to Azure server management services](#)

Feedback

Was this page helpful?

 Yes

 No

Phase 2: Onboarding Azure server management services

Article • 05/23/2023

After you're familiar with the [tools](#) and [planning](#) involved in Azure management services, you're ready for the second phase. Phase 2 provides step-by-step guidance for onboarding these services for use with your Azure resources. Start by evaluating this onboarding process before adopting it broadly in your environment.

ⓘ Note

The automation approaches discussed in later sections of this guidance are meant for deployments that don't already have servers deployed to the cloud. They require that you have the Owner role on a subscription to create all the required resources and policies. If you've already created Log Analytics workspaces and Automation accounts, we recommend that you pass these resources in the appropriate parameters when you start the example automation scripts.

Onboarding processes

This section of the guidance covers the following onboarding processes for both virtual machines in Azure and on-premises servers:

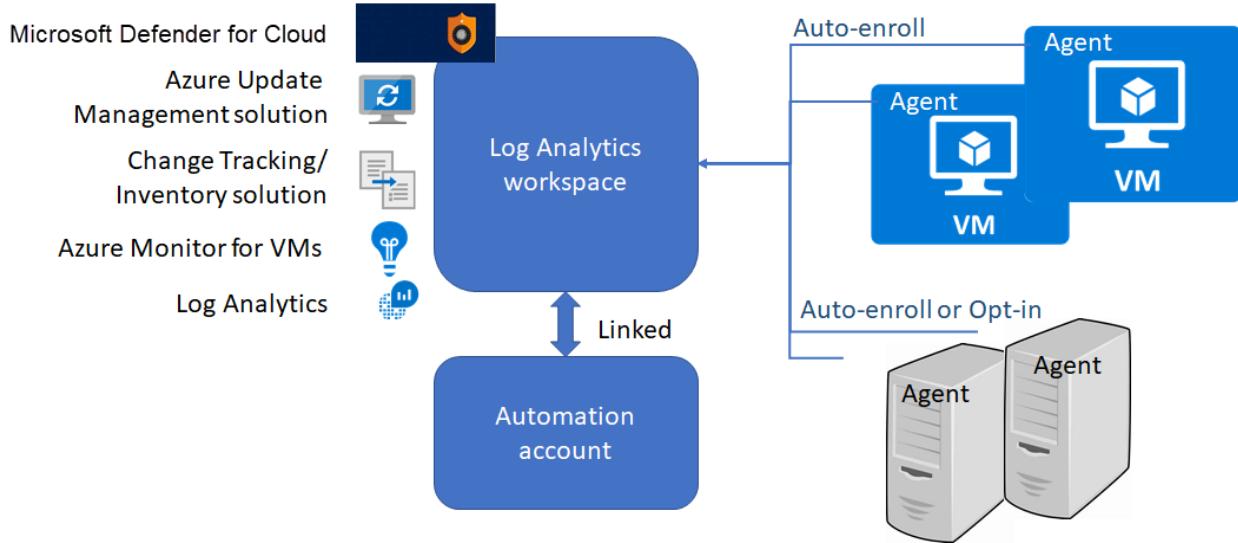
- **Enable management services on a single VM for evaluation by using the portal.**
Use this process to familiarize yourself with the Azure server management services.
- **Configure management services for a subscription by using the portal.** This process helps you configure the Azure environment so that any new VMs that are provisioned will automatically use management services. Use this approach if you prefer the Azure portal experience to scripts and command lines.
- **Configure management services for a subscription by using Azure Automation.**
This process is fully automated. Just create a subscription, and the scripts will configure the environment to use management services for any newly provisioned VM. Use this approach if you're familiar with PowerShell scripts and Azure Resource Manager templates, or if you want to learn to use them.

The procedures for each approach differ.

ⓘ Note

When you use the Azure portal, the sequence of onboarding steps differs from the automated onboarding steps. The portal offers a simpler onboarding experience.

The following diagram shows the recommended deployment model for management services:



As shown in the preceding diagram, the Log Analytics agent has two configurations for on-premises servers:

- **Auto-enroll:** When the Log Analytics agent is installed on a server and configured to connect to a workspace, the solutions that are enabled on that workspace are applied to the server automatically.
- **Opt-in:** Even if the agent is installed and connected to the workspace, the solution isn't applied unless it's added to the server's scope configuration in the workspace.

Tip

Consider using [Azure Automanage machine best practices](#). This service makes it simple to discover, onboard, and configure certain services in Azure that would benefit your virtual machines and assist with [onboarding at scale](#).

Next steps

Learn how to onboard a single VM by using the portal to evaluate the onboarding process.

[Onboard a single Azure VM for evaluation](#)

Enable server management services on a single VM for evaluation

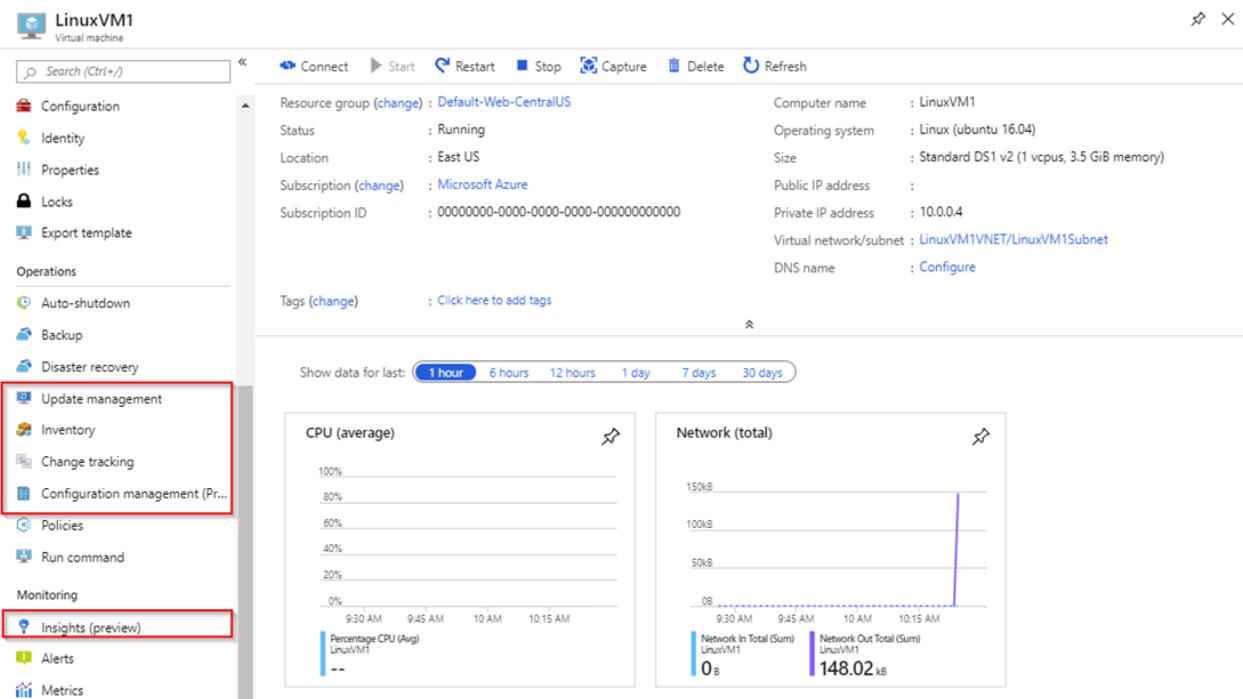
Article • 12/01/2022

Learn how to enable server management services on a single VM for evaluation.

ⓘ Note

Create the required **Log Analytics workspace** and **Azure Automation account** before you implement Azure management services on a VM.

It's simple to onboard Azure server management services to individual virtual machines in the Azure portal. You can familiarize yourself with these services before you onboard them. When you select a VM instance, all the solutions on the list of **management tools and services** appear on the **Operations** or **Monitoring** menu. You select a solution and follow the wizard to onboard it.



Related resources

For more information about how to onboard these solutions to individual VMs, see:

- [Onboard the Update Management solution and the Change Tracking and Inventory solution for a VM in Azure](#)
- [Onboard Azure Monitor for VMs](#)

Next steps

Learn how to use Azure Policy to onboard Azure VMs at scale.

[Configure Azure management services for a subscription](#)

Configure Azure server management services at scale

Article • 12/01/2022

You must complete these two tasks to onboard Azure server management services to your servers:

- Deploy service agents to your servers.
- Enable the management solutions.

This article covers the three processes that are necessary to complete these tasks:

1. Deploy the required agents to Azure VMs by using Azure Policy.
2. Deploy the required agents to on-premises servers.
3. Enable and configuring the solutions.

ⓘ Note

Create the required [Log Analytics workspace](#) and [Azure Automation account](#) before you onboard virtual machines to Azure server management services.

Use Azure Policy to deploy extensions to Azure VMs

All the management solutions that are discussed in [Azure management tools and services](#) require that the Log Analytics agent is installed on virtual machines in Azure as well as on-premises servers. You can onboard your Azure VMs at scale by using Azure Policy. Assign policy to ensure that the agent is installed on your Azure VMs and connected to the correct Log Analytics workspace.

Azure Policy has a [built-in policy initiative](#) that includes the Log Analytics agent and the [Microsoft Dependency Agent](#), which is required by Azure Monitor for VMs.

ⓘ Note

For more information about various agents for monitoring Azure, see [Overview of the Azure monitoring agents](#).

Assign policies

To assign the policies that described in the previous section:

1. In the Azure portal, go to **Policy > Assignments > Assign initiative**.

The screenshot shows the 'Policy - Assignments' blade in the Azure portal. On the left, there's a navigation menu with 'Assignments' selected. At the top right, there are buttons for 'Assign initiative', 'Assign policy', and 'Refresh'. Below these are filters for 'Scope' (set to 'It Azure/TestSubMoveVM'), 'Definition type' (set to 'All definition types'), and a search bar. The main area displays three counts: 'Total Assignments' (5), 'Initiative Assignments' (2), and 'Policy Assignments' (3). A table lists one assignment: 'ASC Default (subscription: 147a22e9-2356-4e5...' with scope 'Microsoft Azure', type 'Initiative', and 70 policies assigned.

2. On the **Assign policy** page, set the **Scope** by selecting the ellipsis (...) and then selecting either a management group or subscription. Optionally, select a resource group. Then choose **Select** at the bottom of the **Scope** page. The scope determines which resources or group of resources the policy is assigned to.
3. Select the ellipsis (...) next to **Policy definition** to open the list of available definitions. To filter the initiative definitions, enter **Azure Monitor** in the **Search** box:

The screenshot shows a search result for '[Preview]: Enable Azure Monitor for VMs' under the 'Built-in' category. The description states: 'Enable Azure Monitor for the Virtual Machines (VMs) in the specified scope (Management group, Subscription or resource group). Takes Log Analytics workspace as parameter.'

4. The **Assignment name** is automatically populated with the policy name that you selected, but you can change it. You can also add an optional description to provide more information about this policy assignment. The **Assigned by** field is automatically filled based on who is signed in. This field is optional, and it supports custom values.
5. For this policy, select **Log Analytics workspace** for the Log Analytics agent to associate.

PARAMETERS

* Log Analytics workspace i

Click '...' to change the subscription for the parameter.



6. Select the **Managed Identity location** checkbox. If this policy is of the type **DeployIfNotExists**, a managed identity will be required to deploy the policy. In the portal, the account will be created as indicated by the checkbox selection.

7. Select Assign.

After you complete the wizard, the policy assignment will be deployed to the environment. It can take up to 30 minutes for the policy to take effect. To test it, create new VMs after 30 minutes, and check if the Log Analytics agent is enabled on the VM by default.

Install agents on on-premises servers

ⓘ Note

Create the required **Log Analytics workspace** and **Azure Automation account** before you onboard Azure server management services to servers.

For on-premises servers, you need to download and install the [Log Analytics agent](#) and [the Microsoft Dependency Agent](#) manually and configure them to connect to the correct workspace. You must specify the workspace ID and key information. To get that information, go to your Log Analytics workspace in the Azure portal, then select **Settings > Advanced settings**.

The screenshot shows the 'Advanced settings' page for a Log Analytics workspace named 'contosofinancelogs'. The left sidebar lists 'Connected Sources', 'Data', and 'Computer Groups'. The main area shows 'Windows Servers' selected under 'Connected Sources'. A red box highlights the 'WORKSPACE ID' field, which contains '20906715-2550-4054-b40f-19e85e712904'. Below it is the 'PRIMARY KEY' field, containing a long alphanumeric string. There are 'Regenerate' buttons for both fields. The right side of the page shows '0 WINDOWS COMPUTERS CONNECTED' and links to download the 'Windows Agent (64 bit)' and 'Windows Agent (32 bit)'. At the bottom, there's information about the 'OMS Gateway' and a link to 'Download OMS Gateway'.

Enable and configure solutions

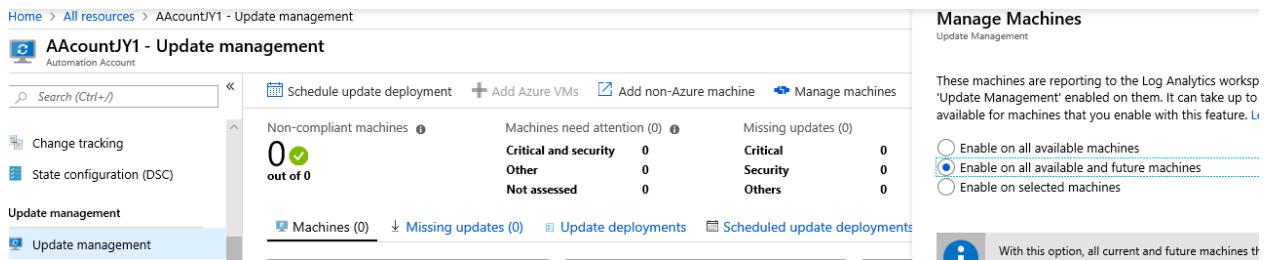
To enable solutions, you need to configure the Log Analytics workspace. Onboarded Azure VMs and on-premises servers will get the solutions from the Log Analytics workspaces that they're connected to.

Update Management

The Update Management solution and the Change Tracking and Inventory solution require both a Log Analytics workspace and an Azure Automation account. To ensure that these resources are properly configured, we recommend that you onboard through your Automation account. For more information, see [Onboard the Update Management solution and the Change Tracking and Inventory solution](#).

We recommend that you enable the Update Management solution for all servers. Update Management is free for Azure VMs and on-premises servers. If you enable Update Management through your Automation account, a [scope configuration](#) is created in the workspace. Manually update the scope to include machines that are covered by the Update Management solution.

To cover your existing servers as well as future servers, you need to remove the scope configuration. To do this, view your Automation account in the Azure portal. Select **Update Management > Manage machine > Enable on all available and future machines**. This setting allows all Azure VMs that are connected to the workspace to use Update Management.



The screenshot shows the Azure portal interface for an Automation account named 'AAccountJY1'. The left sidebar has 'Update management' selected under the 'Automation Account' category. The main content area displays the 'Update management' blade. It includes a summary table:

Non-compliant machines	Machines need attention	Missing updates
0 out of 0	Critical and security: 0 Other: 0 Not assessed: 0	Critical: 0 Security: 0 Others: 0

Below the table are links for 'Machines (0)', 'Missing updates (0)', 'Update deployments', and 'Scheduled update deployments'. To the right, a 'Manage Machines' panel is open, titled 'Update Management'. It contains a note about machines reporting to the workspace and three enablement options:

- Enable on all available machines
- Enable on all available and future machines
- Enable on selected machines

A note at the bottom right says: 'With this option, all current and future machines th'.

Change Tracking and Inventory solutions

To onboard the Change Tracking and Inventory solutions, follow the same steps as for Update Management. For more information about how to onboard these solutions from your Automation account, see [Onboard the Update Management solution and the Change Tracking and Inventory solution](#).

The Change Tracking and Inventory solution is free for Azure VMs and costs \$6 per node per month for on-premises servers. This cost covers change tracking, inventory, and Desired State Configuration. If you want to enroll only specific on-premises servers, you can opt in those servers. We recommend that you onboard all your production servers.

Opt in via the Azure portal

1. Go to the Automation account that has Change Tracking and Inventory enabled.
2. Select **Change tracking**.
3. Select **Manage machines** in the upper-right pane.
4. Select **Enable on selected machines**. Then select **Add** next to the machine name.
5. Select **Enable** to enable the solution for those machines.

The screenshot shows the Azure portal interface for managing machines. On the left, the navigation menu includes 'Automation Account', 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'Configuration Management' (with 'Inventory' selected), 'Update management', 'Process Automation' (with 'Runbooks' selected), 'Jobs', 'Runbooks gallery', 'Hybrid worker groups', 'Watcher tasks', and 'Shared Resources'. In the center, under 'Change tracking', it says '2 machines do not have 'Change Tracking and Inventory' enabled. Click to manage machine'. Below this are sections for 'Events', 'Daemons', 'Files', 'Registry', 'Software', and 'Windows Services', all showing '0' changes. At the bottom, there are tabs for 'Changes (0)' and 'Events (0)'. On the right, the 'Manage Machines' dialog is open. It shows a summary: 'These machines are reporting to the Log Analytics workspace 'workspace-f540e10b-6fc9-486d-8168-91db8986c2ca-eus'', and options to 'Enable on all available machines', 'Enable on all available and future machines', and 'Enable on selected machines' (which is selected). Below this, a note states: 'For non-Azure machines that you enable with 'Change Tracking and Inventory' there may be additional monthly charge. Learn more about pricing'. The 'AVAILABLE MACHINES' section lists 'testvm' and 'testwin' with 'add' buttons. The 'SELECTED MACHINES' section lists 'testwin' with a 'remove' button. At the bottom are 'Enable' and 'Cancel' buttons.

Opt in by using saved searches

Alternatively, you can configure the scope configuration to opt in on-premises servers. Scope configuration uses saved searches.

To create or modify the saved search, follow these steps:

1. Go to the Log Analytics workspace that is linked to the Automation account that you configured in the preceding steps.
2. Under **General**, select **Saved searches**.
3. In the **Filter** box, enter **Change Tracking** to filter the list of saved searches. In the results, select **MicrosoftDefaultComputerGroup**.
4. Enter the computer name or the VMUUID to include the computers that you want to opt in for Change Tracking and Inventory.

The screenshot shows the Kusto query editor. At the top, it says 'Kusto'. Below that, a 'Heartbeat' query is displayed:

```
| where AzureEnvironment=~"Azure" or Computer in~ ("list of the on-premises
```

```
server names", "server1")
| distinct Computer
```

ⓘ Note

The server name must exactly match the value in the expression, and it shouldn't contain a domain name suffix.

1. Select **Save**. By default, the scope configuration is linked to the **MicrosoftDefaultComputerGroup** saved search. It will be automatically updated.

Azure activity log

[Azure activity log](#) is also part of Azure Monitor. It provides insight into subscription-level events that occur in Azure.

To implement this solution:

1. In the Azure portal, open **All services**, then select **Management + Governance > Solutions**.
2. In the **Solutions** view, select **Add**.
3. Search for **Activity Log Analytics** and select it.
4. Select **Create**.

You need to specify the **Workspace name** of the workspace that you created in the previous section where the solution is enabled.

Azure Log Analytics Agent Health

The Azure Log Analytics Agent Health solution reports on the health, performance, and availability of your Windows and Linux servers.

To implement this solution:

1. In the Azure portal, open **All services**, then select **Management + Governance > Solutions**.
2. In the **Solutions** view, select **Add**.
3. Search for **Azure Log Analytics agent health** and select it.
4. Select **Create**.

You need to specify the **Workspace name** of the workspace that you created in the previous section where the solution is enabled.

After creation is complete, the workspace resource instance displays **AgentHealthAssessment** when you select **View > Solutions**.

Antimalware assessment

The antimalware assessment solution helps you identify servers that are infected or at increased risk of infection by malware.

To implement this solution:

1. In the Azure portal, open **All services**, select **Management + Governance > Solutions**.
2. In the **Solutions** view, select **Add**.
3. Search for and then select **Antimalware Assessment**.
4. Select **Create**.

You need to specify the **Workspace name** of the workspace that you created in the previous section where the solution is enabled.

After creation is complete, the workspace resource instance displays **AntiMalware** when you select **View > Solutions**.

Azure Monitor for VMs

You can enable [Azure Monitor for VMs](#) through the view page for the VM instance, as described in [Enable management services on a single VM for evaluation](#). You shouldn't enable solutions directly from the **Solutions** page as you do for the other solutions that are described in this article. For large-scale deployments, it may be easier to use [automation](#) to enable the correct solutions in the workspace.

Microsoft Defender for Cloud

We recommend that you onboard all your servers at least to the Free tier of Microsoft Defender for Cloud. This option provides basic security assessments and actionable security recommendations for your environment. The Standard tier provides additional benefits. For more information, see [Microsoft Defender for Cloud pricing](#).

To enable the Free tier of Microsoft Defender for Cloud, follow these steps:

1. Go to the [Defender for Cloud](#) portal page.
2. Under **POLICY & COMPLIANCE**, select **Security policy**.

3. Find the Log Analytics workspace resource that you created in the pane on the right side.
4. Select **Edit settings** for that workspace.
5. Select **Pricing tier**.
6. Choose the **Free** option.
7. Select **Save**.

Next steps

Learn how to use automation to onboard servers and create alerts.

[Automate onboarding and alert configuration](#)

Automate onboarding

Article • 12/01/2022

To improve the efficiency of deploying Azure server management services, consider automating deployment as discussed in previous sections of this guidance. The script and the example templates provided in the following sections are starting points for developing your own automation of onboarding processes.

This guidance is supported by a [GitHub repository of sample code](#). The repository provides example scripts and Azure Resource Manager templates to help you automate the deployment of Azure server management services.

The sample files illustrate how to use Azure PowerShell cmdlets to automate the following tasks:

- Create a [Log Analytics workspace](#). (Or, use an existing workspace if it meets the requirements. For details, see [Workspace planning](#).)
- Create an Azure Automation account, or use an existing account that meets the requirements. For more information, see [Workspace planning](#).
- Link the Automation account and the Log Analytics workspace. This step isn't required if you're onboarding by using the Azure portal.
- Enable the Update Management solution and the Change Tracking and Inventory solution for the workspace.
- Onboard Azure VMs by using Azure Policy. A policy installs the Log Analytics agent and the Microsoft Dependency Agent on the Azure VMs.
- Auto-enable Azure Backup for VMs using [Azure Policy](#)
- Onboard on-premises servers by installing the Log Analytics agent on them.

The files described in the following table are used in this sample. You can customize them to support your own deployment scenarios.

File name	Description
New-AMSDeployment.ps1	The main, orchestrating script that automates onboarding. It creates resource groups, and location, workspace, and Automation accounts, if they don't exist already. This PowerShell script requires an existing subscription.

File name	Description
<code>Workspace-AutomationAccount.json</code>	A Resource Manager template that deploys the workspace and Automation account resources.
<code>WorkspaceSolutions.json</code>	A Resource Manager template that enables the solutions you want in the Log Analytics workspace.
<code>ScopeConfig.json</code>	A Resource Manager template that uses the opt-in model for on-premises servers with the Change Tracking and Inventory solution. Using the opt-in model is optional.
<code>Enable-VMInsightsPerfCounters.ps1</code>	A PowerShell script that enables Azure Monitor for VMs and configures performance counters.
<code>ChangeTracking-FileList.json</code>	A Resource Manager template that defines the list of files that will be monitored by change tracking.

Use the following command to run `New-AMSDeployment.ps1`:

PowerShell

```
.\New-AMSDeployment.ps1 -SubscriptionName '{Subscription Name}' -
WorkspaceName '{Workspace Name}' -WorkspaceLocation '{Azure Location}' -
AutomationAccountName {Account Name} -AutomationAccountLocation {Account
Location}
```

Next steps

Learn how to set up basic alerts to notify your team of key management events and issues.

[Set up basic alerts](#)

Set up basic alerts

Article • 12/01/2022

A key part of managing resources is getting notified when problems occur. Alerts proactively notify you of critical conditions, based on triggers from metrics, logs, or service-health issues. As part of onboarding the Azure server management services, you can set up alerts and notifications that help keep your IT teams aware of any problems.

Azure Monitor alerts

Azure Monitor offers [alerting](#) capabilities to notify you, via email or messaging, when things go wrong. These capabilities are based on a common data-monitoring platform that includes logs and metrics generated by your servers and other resources. By using a common set of tools in Azure Monitor, you can analyze data that's combined from multiple resources and use it to trigger alerts. These triggers can include:

- Metric values.
- Log search queries.
- Activity log events.
- The health of the underlying Azure platform.
- Tests for website availability.

See the [list of Azure Monitor data sources](#) for a more detailed description of the sources of monitoring data that this service collects.

For details about manually creating and managing alerts by using the Azure portal, see the [Azure Monitor documentation](#).

Automated deployment of recommended alerts

In this guide, we recommend that you create a set of 15 alerts for basic infrastructure monitoring. Find the deployment scripts in the [Alert Toolkit](#) GitHub repository.

This package creates alerts for:

- Low disk space
- Low available memory
- High CPU use
- Unexpected shutdowns

- Corrupted file systems
- Common hardware failures

The package uses HPE server hardware as an example. Change the settings in the associated configuration file to reflect your OEM hardware. You can also add more performance counters to the configuration file. To deploy the package, run the `New-CoreAlerts.ps1` file.

Next steps

Learn about operations and security mechanisms that support your ongoing operations.

[Ongoing management and security](#)

Phase 3: Ongoing management and security

Article • 07/16/2024

After you've onboarded Azure server management services, you'll need to focus on the operations and security configurations that will support your ongoing operations. We'll start with securing your environment by reviewing Microsoft Defender for Cloud. We'll then configure policies to keep your servers in compliance and automate common tasks. This section covers the following topics:

- [Enable the machine configuration policy](#): Use the Azure Policy machine configuration feature to audit the settings in a virtual machine. For example, you can check whether any certificates are about to expire.
- [Track and alert on critical changes](#): When you're troubleshooting, the first question to consider is, "What has changed?" In this article, you'll learn how to track changes and create alerts to proactively monitor critical components.
- [Create update schedules](#): Schedule the installation of updates to ensure that all your servers have the latest ones.
- [Common Azure Policy examples](#): This article provides examples of common management policies.
- [Protect cloud workloads](#): See how Microsoft Defender for Cloud can help you protect your cloud workloads.

Next steps

Learn how to [enable the Azure Policy machine configuration](#) feature.

[Machine configuration policy](#)

Feedback

Was this page helpful?

 Yes

 No

Azure Policy machine configuration extension

Article • 07/17/2024

You can use the [Azure Policy machine configuration extension](#) to audit the configuration settings of a virtual machine. Machine configuration supports Azure VMs natively. Physical servers and non-Azure virtual servers are supported with [Azure Arc-enabled servers](#), [Azure Arc-enabled VMware vSphere](#), or [Azure Arc-enabled System Center Virtual Machine Manager](#).

To find the list of machine configuration policies, search for *machine configuration* on the Azure Policy portal page, or run this cmdlet in a PowerShell window to find the list:

PowerShell

```
Get-AzPolicySetDefinition | Where-Object {$_.Properties.metadata.category -eq "Machine Configuration"}
```

ⓘ Note

Machine configuration functionality is regularly updated to support additional policy sets. Check for new supported policies periodically and evaluate whether they'll be useful.

Deployment

Use the following example PowerShell script to deploy these policies to:

- Verify that password security settings in Windows and Linux computers are set correctly.
- Verify that certificates aren't close to expiration on Windows VMs.

Before you run this script, use the [Connect-AzAccount](#) cmdlet to sign in. When you run the script, you must provide the name of the subscription that you want to apply the policies to.

PowerShell

```
# Assign machine configuration policy.
```

```
param (
    [Parameter(Mandatory=$true)]
    [string] $SubscriptionName
)

$Subscription = Get-AzSubscription -SubscriptionName $SubscriptionName
$scope = "/subscriptions/" + $Subscription.Id

$PasswordPolicy = Get-AzPolicySetDefinition -Name "3fa7cbf5-c0a4-4a59-85a5-cca4d996d5a6"
$CertExpirePolicy = Get-AzPolicySetDefinition -Name "b6f5e05c-0aaa-4337-8dd4-357c399d12ae"

New-AzPolicyAssignment -Name "PasswordPolicy" -DisplayName "[Preview]: Audit that password security settings are set correctly inside Linux and Windows machines" -Scope $scope -PolicySetDefinition $PasswordPolicy -AssignIdentity -Location eastus

New-AzPolicyAssignment -Name "CertExpirePolicy" -DisplayName "[Preview]: Audit that certificates are not expiring on Windows VMs" -Scope $scope -PolicySetDefinition $CertExpirePolicy -AssignIdentity -Location eastus
```

Next steps

Learn how to [enable change tracking and alerting](#) for critical file, service, software, and registry changes.

[Enable tracking and alerting for critical changes](#)

Feedback

Was this page helpful?

 Yes

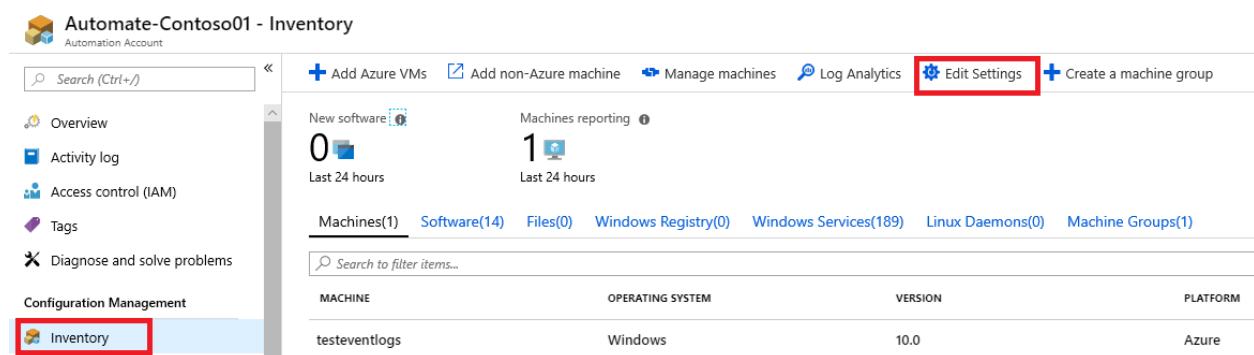
 No

Enable tracking and alerting for critical changes

Article • 12/01/2022

Azure Change Tracking and Inventory provide alerts on the configuration state of your hybrid environment and changes to that environment. It can report critical file, service, software, and registry changes that might affect your deployed servers.

By default, the Azure Automation inventory service doesn't monitor files or registry settings. The solution does provide a list of registry keys that we recommend for monitoring. To see this list, go to your Azure Automation account in the Azure portal, then select **Inventory > Edit settings**.



The screenshot shows the Azure Automation Inventory interface. At the top, there's a navigation bar with links for 'Add Azure VMs', 'Add non-Azure machine', 'Manage machines', 'Log Analytics', 'Edit Settings' (which is highlighted with a red box), and 'Create a machine group'. Below the navigation bar, there are summary statistics: 'New software' (0 last 24 hours), 'Machines reporting' (1 last 24 hours), 'Machines(1)', 'Software(14)', 'Files(0)', 'Windows Registry(0)', 'Windows Services(189)', 'Linux Daemons(0)', and 'Machine Groups(1)'. A search bar labeled 'Search to filter items...' is present. The main table lists one machine: 'testeventlogs' with 'Windows' operating system, '10.0' version, and 'Azure' platform. On the left sidebar, there are links for 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'Configuration Management', and 'Inventory' (which is highlighted with a red box). A 'Search (Ctrl+J)' input field is at the top left of the sidebar.

For more information about each registry key, see [Registry key change tracking](#). Select any key to evaluate and then enable it. The setting is applied to all VMs that are enabled in the current workspace.

You can also use the service to track critical file changes. For example, you might want to track the `c:\windows\system32\drivers\etc\hosts` file because the OS uses it to map host names to IP addresses. Changes to this file could cause connectivity problems or redirect traffic to dangerous websites.

To enable file-content tracking for the hosts file, follow the steps in [Enable file content tracking](#).

You can also add an alert for changes to files that you're tracking. For example, you might want to set an alert for changes to the hosts file. To do this, select **Log Analytics** on the command bar or **Log Search** for the linked Log Analytics workspace. In Log Analytics, use the following query to search for changes to the hosts file:

```
Kusto

ConfigurationChange | where FieldsChanged contains "FileContentChecksum" and
FileSystemPath contains "hosts"
```

The screenshot shows the Azure Monitor Log Analytics search interface. At the top, there are buttons for 'Run' (highlighted in blue), 'Save', 'Copy link', 'Export', 'New alert rule' (highlighted in blue), and 'Pin'. Below the buttons, a time range selector shows 'Last 24 hours'. The main area contains a search bar with the query: ConfigurationChange | where FieldsChanged contains "FileContentChecksum" and FileSystemPath contains "hosts".

This query searches for changes to the contents of files that have a path that contains the word `hosts`. You can also search for a specific file by changing the path parameter. (For example: `FileSystemPath == "c:\\windows\\system32\\drivers\\etc\\hosts"`.)

After the query returns the results, select **New alert rule** to open the alert-rule editor. You can also get to this editor via Azure Monitor in the Azure portal.

In the alert-rule editor, review the query and change the alert logic if you need to. In this case, we want the alert to be raised if any changes are detected on any machine in the environment.

The screenshot shows the Azure Monitor Alert Rule Editor. The 'Search query' field contains the query: ConfigurationChange | where FieldsChanged contains "FileContentChecksum" and FileSystemPath contains "hosts". Below the query, it says 'Query to be executed : ConfigurationChange | where FieldsChanged contains "FileContentChecksum" and FileSystemPath contains "hosts" &| count' and 'For time window : 1/4/2019, 4:17:49 PM - 1/4/2019, 4:22:49 PM'. The 'Alert logic' section includes fields for 'Based on' (set to 'Number of results'), 'Condition' (set to 'Greater than'), and 'Threshold' (set to '1'). The 'Condition preview' section states 'Whenever the custom log search is greater than 1 count'. The 'Evaluated based on' section includes fields for 'Period (in minutes)' (set to '5') and 'Frequency (in minutes)' (set to '5').

After you set the condition logic, you can assign action groups to perform actions in response to the alert. In this example, when the alert is raised, emails are sent and an ITSM ticket is created. You can take many other useful actions, like triggering an Azure function, an Azure Automation runbook, a webhook, or a logic app.

The screenshot shows the configuration of an alert rule in the Microsoft Azure portal. At the top, it displays the resource 'contosomarketingworkspace' and its hierarchy under 'contoso IT - demo > contosoautomation'. The 'CONDITION' section contains a single rule: 'Whenever the Custom log search is Greater than 1 count', which costs \$1.50. A note indicates that only two metrics signals or one log search signal or one activity log signal can be configured per alert rule. The 'ACTION GROUPS' section lists two actions: 'Create Ticket' (1 ITSM) and 'Send Email' (2 Email(s)). Buttons for 'Select existing' and 'Create New' are available at the bottom.

After you've set all the parameters and logic, apply the alert to the environment.

Tracking and alerting examples

This section shows other common scenarios for tracking and alerting that you might want to use.

Driver file changed

Use the following query to detect if driver files are changed, added, or removed. It's useful for tracking changes to critical system files.

Kusto

```
ConfigurationChange | where ConfigChangeType == "Files" and FileSystemPath  
contains " c:\\windows\\system32\\drivers\\"
```

Specific service stopped

Use the following query to track changes to system-critical services.

Kusto

```
ConfigurationChange | where SvcState == "Stopped" and SvcName contains  
"w3svc"
```

New software installed

Use the following query for environments that need to lock down software configurations.

Kusto

```
ConfigurationChange | where ConfigChangeType == "Software" and  
ChangeCategory == "Added"
```

Specific software version is or isn't installed on a machine

Use the following query to assess security. This query references `ConfigurationData`, which contains the logs for inventory and provides the last-reported configuration state, not changes.

Kusto

```
ConfigurationData | where SoftwareName contains "Monitoring Agent" and  
CurrentVersion != "8.0.11081.0"
```

Known DLL changed through the registry

Use the following query to detect changes to well-known registry keys.

Kusto

```
ConfigurationChange | where RegistryKey ==  
"HKEY_LOCAL_MACHINE\\System\\CurrentControlSet\\Control\\Session  
Manager\\KnownDlls"
```

Next steps

Learn how Azure Automation can [create update schedules](#) to manage updates for your servers.

[Create update schedules](#)

Create update schedules

Article • 12/01/2022

You can manage update schedules by using the Azure portal or the new PowerShell cmdlet modules.

To create an update schedule via the Azure portal, see [Schedule an update deployment](#).

The `Az.Automation` module now supports configuring Update Management by using Azure PowerShell. The `New-AzAutomationUpdateManagementAzureQuery` cmdlet allows you to use tags, location, and saved searches to configure update schedules for a flexible group of machines.

Example script

The example script in this section illustrates the use of tagging and querying to create dynamic groups of machines that you can apply update schedules to. It performs the following actions. You can refer to the implementations of the specific actions when you create your own scripts.

- Creates an Azure Automation update schedule that runs every Saturday at 8:00 AM.
- Creates a query for any machines that match these criteria:
 - Deployed in the `westus`, `eastus`, or `eastus2` Azure location.
 - Has an `Owner` tag applied with a value set to `JaneSmith`.
 - Has a `Production` tag applied with a value set to `true`.
- Applies the update schedule to the queried machines and sets a two-hour update window.

Before you run the example script, you'll need to sign in by using the [Connect-AzAccount](#) cmdlet. When you start the script, provide the following information:

- The target subscription ID
- The target resource group
- Your Log Analytics workspace name
- Your Azure Automation account name

```
PowerShell

<#
 .SYNOPSIS
```

```

    This script orchestrates the deployment of the solutions and the
agents.

.Parameter SubscriptionName
.Parameter WorkspaceName
.Parameter AutomationAccountName
.Parameter ResourceGroupName

#>

param (
    [Parameter(Mandatory=$true)]
    [string] $SubscriptionId,

    [Parameter(Mandatory=$true)]
    [string] $ResourceGroupName,

    [Parameter(Mandatory=$true)]
    [string] $WorkspaceName,

    [Parameter(Mandatory=$true)]
    [string] $AutomationAccountName,

    [Parameter(Mandatory=$false)]
    [string] $scheduleName = "SaturdayCriticalSecurity"
)

Import-Module Az.Automation

$startTime = ([DateTime]::Now).AddMinutes(10)
$schedule = New-AzAutomationSchedule -ResourceGroupName
$ResourceGroupName `-
    -AutomationAccountName $AutomationAccountName `-
    -StartTime $startTime `-
    -Name $scheduleName `-
    -Description "Saturday patches" `-
    -DaysOfWeek Saturday `-
    -WeekInterval 1 `-
    -ForUpdateConfiguration

# Using AzAutomationUpdateManagementAzureQuery to create dynamic groups.

$queryScope = @("/subscriptions/$SubscriptionID/resourceGroups/")

$query1Location =@("westus", "eastus", "eastus2")
$query1FilterOperator = "Any"
$ownerTag = @{ "Owner"= @("JaneSmith") }
$ownerTag.Add("Production", "true")

$DGQuery = New-AzAutomationUpdateManagementAzureQuery -ResourceGroupName
$ResourceGroupName `-
    -AutomationAccountName $AutomationAccountName `-
    -Scope $queryScope `-
    -Tag $ownerTag

$AzureQueries = @($DGQuery)

```

```
$UpdateConfig = New-AzAutomationSoftwareUpdateConfiguration -  
ResourceGroupName $ResourceGroupName `  
-AutomationAccountName $AutomationAccountName `  
-Schedule $schedule `  
-Windows `  
-Duration (New-TimeSpan -Hours 2) `  
-AzureQuery $AzureQueries `  
-IncludedUpdateClassification Security,Critical
```

Next steps

See examples of how to implement [common policies in Azure](#) that can help manage your servers.

[Common policies in Azure](#)

Common Azure Policy examples

Article • 03/06/2023

Azure Policy can help you apply governance to your cloud resources. This service can help you create guardrails that ensure company-wide compliance to governance policy requirements. To create policies, use either the Azure portal or PowerShell cmdlets. This article provides PowerShell cmdlet examples.

ⓘ Note

With Azure Policy, enforcement policies (`DeployIfNotExists`) aren't automatically deployed to existing VMs. Remediation is required to keep VMs in compliance. For more information, see [Remediate noncompliant resources with Azure Policy](#).

Common policy examples

The following sections describe some commonly used policies.

Restrict resource regions

Regulatory and policy compliance often depends on control of the physical location where resources are deployed. You can use a built-in policy to allow users to create resources only in certain allowed Azure regions.

To find this policy in the portal, search for "location" on the policy definition page. Or run this cmdlet to find the policy:

PowerShell

```
Get-AzPolicyDefinition | Where-Object { ($_.Properties.policyType -eq 'BuiltIn') ` -and ($_.Properties.displayName -like '*location*') }
```

The following script shows how to assign the policy. Change the `$SubscriptionID` value to point to the subscription that you want to assign the policy to. Before you run the script, use the [Connect-AzAccount](#) cmdlet to sign in.

PowerShell

```
# Specify the value for $SubscriptionID.
```

```

$SubscriptionID = <subscription ID>
$scope = "/subscriptions/$SubscriptionID"

# Replace the -Name GUID with the policy GUID you want to assign.

$AllowedLocationPolicy = Get-AzPolicyDefinition -Name "e56962a6-4747-49cd-
b67b-bf8b01975c4c"

# Replace the locations with the ones you want to specify.

$policyParam = '{ "listOfAllowedLocations":{"value":["eastus","westus"]}}'
New-AzPolicyAssignment -Name "Allowed Location" -DisplayName "Allowed
locations for resource creation" -Scope $scope -PolicyDefinition
$AllowedLocationPolicy -Location eastus -PolicyParameter $policyParam

```

You can also use this script to apply the other policies that are discussed in this article. Just replace the GUID in the line that sets `$AllowedLocationPolicy` with the GUID of the policy that you want to apply.

Block certain resource types

Another common built-in policy that's used to control costs can also be used to block certain resource types.

To find this policy in the portal, search for "allowed resource types" on the policy definition page. Or run this cmdlet to find the policy:

PowerShell

```

Get-AzPolicyDefinition | Where-Object { ($_.Properties.policyType -eq
"BuiltIn") -and ($_.Properties.displayName -like "*allowed resource types")
}

```

After you identify the policy that you want to use, you can modify the PowerShell sample in the [Restrict resource regions](#) section to assign the policy.

Restrict VM size

Azure offers a wide range of VM sizes to support various workloads. To control your budget, you could create a policy that allows only a subset of VM sizes in your subscriptions.

Deploy antimalware

You can use this policy to deploy the Microsoft Antimalware Extension with a default configuration to VMs that aren't protected by antimalware.

The policy GUID is `2835b622-407b-4114-9198-6f7064cbe0dc`.

The following script shows how to assign the policy. To use the script, change the `$SubscriptionID` value to point to the subscription that you want to assign the policy to.

Before you run the script, use the [Connect-AzAccount](#) cmdlet to sign in.

PowerShell

```
# Specify the value for $SubscriptionID.

$subscriptionID = <subscription ID>
$scope = "/subscriptions/$subscriptionID"

$antimalwarePolicy = Get-AzPolicyDefinition -Name "2835b622-407b-4114-9198-6f7064cbe0dc"

# Replace location "eastus" with the value that you want to use.

New-AzPolicyAssignment -Name "Deploy Antimalware" -DisplayName "Deploy default Microsoft IaaSAntimalware extension for Windows Server" -Scope $scope -PolicyDefinition $antimalwarePolicy -Location eastus -AssignIdentity
```

Next steps

Learn about other server-management tools and services that are available.

[Azure server management tools and services](#)

Azure server management tools and services

Article • 12/03/2024

As is discussed in the [overview](#) of this guidance, the suite of Azure server management services covers these areas:

- Migrate
- Secure
- Protect
- Monitor
- Configure
- Govern

The following sections briefly describe these management areas and provide links to detailed content about the main Azure services that support them.

Migrate

Migration services can help you migrate your workloads into Azure. To provide the best guidance, the [Azure Migrate](#) service starts by measuring on-premises server performance and assessing suitability for migration. After Azure Migrate completes the assessment, you can use [Azure Site Recovery](#) and [Azure Database Migration Service](#) to migrate your on-premises machines to Azure.

Secure

[Microsoft Defender for Cloud](#) is a comprehensive security management application. By onboarding to Defender for Cloud, you can quickly get an assessment of the security and regulatory compliance status of your environment. For instructions on onboarding your servers to Defender for Cloud, see [Configure Azure management services for a subscription](#).

Protect

To protect your data, you need to plan for backup, high availability, encryption, authorization, and related operational issues. These topics are covered extensively online, so here we'll focus on building a business continuity and disaster recovery

(BCDR) plan. We'll include references to documentation that describes in detail how to implement and deploy this type of plan.

When you build data-protection strategies, first consider breaking down your workload applications into their different tiers. This approach helps because each tier typically requires its own unique protection plan. To learn more about designing applications to be resilient, see [Designing resilient applications for Azure](#).

The most basic data protection is backup. To speed up the recovery process if servers are lost, back up not just data but also server configurations. Backup is an effective mechanism to handle accidental data deletion and ransomware attacks. [Azure Backup](#) can help you protect your data on Azure and on-premises servers running Windows or Linux. For details about what Azure Backup can do and for how-to guides, see the [Azure Backup service overview](#).

If a workload requires real-time business continuity for hardware failures or datacenter outage, consider using data replication. [Azure Site Recovery](#) provides continuous replication of your VMs, a solution that provides bare-minimum data loss. Site Recovery also supports several replication scenarios, such as replication:

- Of Azure VMs between two Azure regions.
- Between servers on-premises.
- Between on-premises servers and Azure.

For more information, see the [complete Azure Site Recovery replication matrix](#).

For your file-server data, another service to consider is [Azure File Sync](#). This service helps you centralize your organization's file shares in Azure Files, while preserving the flexibility, performance, and compatibility of an on-premises file server. To use this service, follow the instructions for deploying Azure File Sync.

An efficient and easy-to-use web-based user interface is [Azure Business Continuity Center](#). This console enables you to manage your backup and disaster recovery at scale from a single place across various environments and solutions.

Monitor

[Azure Monitor](#) provides a view into various resources, like applications, containers, and virtual machines. It also collects data from several sources:

- [Azure Monitor for VMs](#) provides an in-depth view of VM health, performance trends, and dependencies. The service monitors the health of the operating

systems of your Azure virtual machines, virtual-machine scale sets, and machines in your on-premises environment.

- [Log Analytics](#) is a feature of Azure Monitor. Its role is central to the overall Azure management story. It serves as the data store for log analysis and for many other Azure services. It offers a rich query language and an analytics engine that provides insights into the operation of your applications and resources.
- [Azure activity log](#) is also a feature of Azure Monitor. It provides insight into subscription-level events that occur in Azure.

Configure

Several services fit into this category. They can help you to:

- Automate operational tasks.
- Manage server configurations.
- Measure update compliance.
- Schedule updates.
- Detect changes to your servers.

These services are essential to supporting ongoing operations:

- [Azure Update Manager](#) automates the deployment of patches across your environment, including deployment to operating-system instances running outside of Azure. It supports both Windows and Linux operating systems, and tracks key OS vulnerabilities and nonconformance caused by missing patches.
- [Change Tracking and Inventory](#) provides insight into the software that's running in your environment, and highlights any changes that have occurred.
- [Azure Automation](#) lets you run Python and PowerShell scripts or runbooks to automate tasks across your environment. When you use Azure Automation with the [Hybrid Runbook Worker](#), you can extend your runbooks to your on-premises resources as well.
- [Azure Automation State Configuration](#) enables you to push PowerShell Desired State Configuration (DSC) configurations directly from Azure. DSC also lets you monitor and preserve configurations for guest operating systems and workloads.

Govern

Adopting and moving to the cloud creates new management challenges. It requires a different mindset as you shift from an operational management burden to monitoring and governance. The Cloud Adoption Framework starts with [governance](#). The framework

explains how to migrate to the cloud, what the journey will look like, and who should be involved.

The governance design for standard organizations often differs from governance design for complex enterprises. To learn more about governance best practices for a standard organization, see the [standard enterprise governance guide](#). To learn more about governance best practices for a complex enterprise, see the [governance guide for complex enterprises](#).

Billing information

To learn about pricing for Azure management services, go to these pages:

- [Azure Site Recovery](#)
- [Azure Migrate](#)
- [Azure Backup](#)
- [Azure Monitor](#)
- [Microsoft Defender for Cloud](#)
- [Azure Automation](#), including:
 - Desired State Configuration
 - Change Tracking and Inventory solution
- [Azure Update Manager](#)
- [Azure Policy](#)
- [Azure File Sync service](#)

Note

The Azure Update Management solution is free, but there's a small cost related to data ingestion. As a rule of thumb, the first 5 gigabytes (GB) per month of data ingestion are free. We generally observe that each machine uses about 25 MB per month. So, about 200 machines per month are covered for free. For more servers, multiply the number of additional servers by 25 MB per month. Then, multiply the result by the storage price for the additional storage that you need. For information about costs, see [Azure Storage pricing](#). Each additional server typically has a nominal impact on cost.

Feedback

Was this page helpful?

 Yes

 No

Monitor a cloud environment

Article • 11/12/2024

You need observability of your cloud environment to help ensure that your workloads run smoothly, whether you're a business owner, platform owner, or application owner. You need to know if:

- Your applications are available and if they perform to your customers' expectations.
- You have any security threats that require investigation.
- Your consumption costs are within the expected range.

Monitoring is the process of collecting, analyzing, and acting on telemetry that indicates the health of your platform, resources, and applications. An effective monitoring environment includes your entire cloud estate, which might include resources across multiple clouds and on-premises environments.

Observability is a property of a system that measures how well its internal states can be inferred from its external outputs. You need to deploy services and processes to monitor your cloud environment. And you need to have the ability to observe and understand the behavior of your services that run in the cloud.

Benefits of monitoring

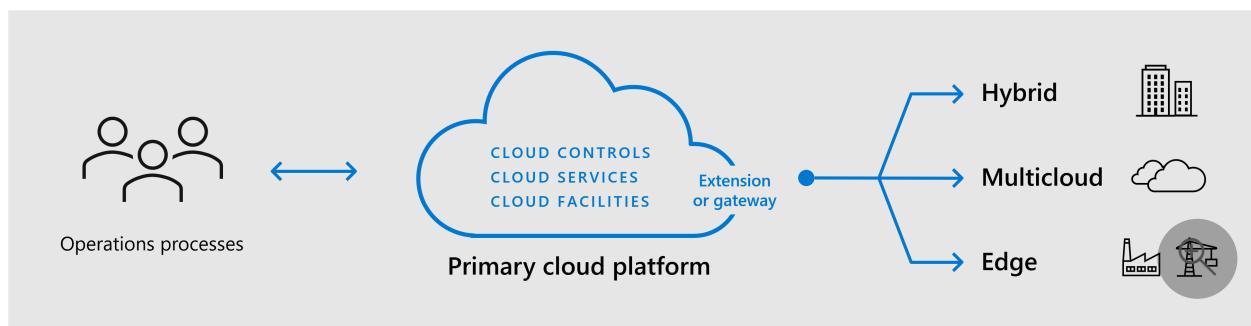
Invest in your monitoring environment to get the following benefits across multiple aspects of your cloud:

- **Availability and performance:** Monitor resources to help ensure that your cloud services and applications are available and perform as expected. To identify and respond to problems before they affect users, track key metrics and configure alert rules.
- **Cost Optimization:** Use monitoring to track resource usage and scale resources according to demand. This approach helps prevent overprovisioned and underused resources, which optimizes cost. Monitoring can also identify and alert you to any cost overruns or unexpected spikes in usage.
- **Compliance:** Use monitoring to maintain logs and records of activities, which help ensure that cloud services comply with policies and regulations. Reports that use this data can assist with regular audits and compliance checks.

- **Security:** Implement continuous monitoring to help detect security threats and vulnerabilities so that you can immediately act to protect data and resources. You can also analyze collected data for threat detection and response.

Monitoring platforms

An effective monitoring strategy includes all platforms in your computing environment. In addition to Azure, you might have on-premises, multicloud, and edge resources. Each resource requires the same levels of monitoring. Follow the [Cloud Adoption Framework for Azure guidance](#), and include monitoring in your [unified operations strategy](#). In this strategy, the primary cloud hosts your monitoring tools and other management tools. The monitoring tools monitor all resources across all platforms.



Types of monitoring

Monitoring is a multifaceted discipline that requires a combination of tools, processes, and practices. The following table breaks down various types of monitoring. Different services and features might provide different combinations of these monitoring types. But a comprehensive monitoring environment includes all of these monitoring types across each of the platforms in your computing environment.

[\[\] Expand table](#)

Type	Description
Infrastructure	Infrastructure monitoring includes the performance and availability of cloud resources, such as virtual machines, storage resources, and networks. This type of monitoring helps ensure that the underlying infrastructure functions optimally, which helps maintain the availability and performance of the applications that rely on it.
Application performance monitoring (APM)	APM monitors the performance and availability of applications that run in the cloud. It tracks metrics such as response times, error rates, and transaction volumes. APM identifies performance bottlenecks and helps ensure that applications meet user expectations.

Type	Description
Database	Database monitoring tracks the performance, availability, and resource consumption of cloud databases. Key metrics include query performance, index usage, and lock status.
Network	Network monitoring tracks the performance and availability of network components in the cloud environment. Metrics include bandwidth usage, latency, and packet loss.
Security	Security monitoring tracks and analyzes security events and vulnerabilities within the cloud environment, including unauthorized access, malware, and compliance violations. Effective security monitoring helps protect sensitive data, ensure compliance with regulatory requirements, and prevent costly security breaches.
Compliance	Compliance monitoring helps ensure that the cloud environment adheres to regulatory and industry standards. It tracks configurations, access controls, and data-handling practices to help ensure compliance with relevant regulations.
Cost	Cost monitoring tracks cloud spending and resource usage to identify cost-saving opportunities and prevent budget overruns. It monitors resource usage, identifies underused resources, and optimizes resource configurations to help reduce costs.

Shared responsibilities

In an on-premises environment, you're responsible for all aspects of monitoring because you own and manage all computing resources. In the cloud, you share this responsibility with your cloud provider. Depending on the type of deployment model that you choose, the responsibilities for monitoring various layers of the cloud stack might transfer from you to your cloud provider.

In an infrastructure as a service (IaaS) deployment, the cloud provider monitors the underlying cloud platform, such as the physical infrastructure and virtualization layer. And you monitor the operating system, applications, and data that run on the virtual machines that you deploy to the cloud platform. When the deployment model moves up the stack, the cloud provider takes on more responsibility to monitor the environment. This responsibility culminates in a software as a service (SaaS) deployment because you transfer monitoring responsibility to the cloud provider for the entire stack, including the application and data.

	On-Premises	IaaS	PaaS	SaaS
Application	■	■	■	□
Data	■	■	■	□
Runtime	■	■	□	□
OS	■	■	□	□
Virtualization	■	□	□	□
Servers	■	□	□	□
Storage	■	□	□	□
Networking	■	□	□	🔍

You might use monitoring tools from the cloud provider to monitor your layers of the stack, but you're responsible for configuring these tools and analyzing the data that they collect. You need to grant access to various members of your organization and create dashboards and alerts to help them distinguish critical information. You might also need to integrate these components with other tools and ticketing systems that your organization uses.

The cloud provider must perform the same types of service for their layers of the stack that you provide to your internal customers. They must continuously monitor the health and performance of the platform that they contract to you. They provide you with dashboards and alerts to proactively notify you of any service problems. Much like your internal customers, you don't need visibility into the intricacies of how the cloud provider monitors their platform, only that they meet the service-level agreements that you contract with them.

Roles and responsibilities

Most enterprise organizations have a centralized operations team that monitors the overall health and performance of the cloud environment.

This team typically:

- Sets the strategies for the overall company.
- Performs centralized configuration of the monitoring environment.
- Delegates permissions to stakeholders in your organization that require access to the monitoring data that's related to their applications and services.

Organizations have multiple roles that maintain the monitoring environment and that require access to monitoring data to perform their job functions. Each role has different requirements to monitor data based on their particular responsibilities. Depending on the size of your organization, you might have multiple individuals that fill each role, or you might have one individual that fills multiple roles.

Individual organizations might distribute responsibilities differently. The following table shows an example of the roles and responsibilities for a typical organization.

 Expand table

Role	Description
Cloud architect	The cloud architect designs and oversees the cloud infrastructure to help ensure that it meets the organization's business goals. The cloud architect focuses on reliability, security, and scalability of the cloud architecture. They require high-level telemetry to get a holistic view of the digital estate. This telemetry includes resource usage metrics, APM metrics, cost and billing insights, and compliance reports.
Platform engineer	The platform engineer builds and manages the platform that developers use to deploy their applications. The platform engineer might create continuous integration and continuous delivery (CI/CD) pipelines, manage cloud infrastructure as code (IaC), and ensure the scalability and reliability of the platform. The platform engineer requires telemetry about the platform's operational status. This telemetry includes container performance metrics, orchestration logs, IaC validation, and service availability.
System administrator	The system administrator manages and maintains servers, operating systems, and other infrastructure components in the cloud. They perform backups, troubleshoot problems, and ensure that systems are up to date. The system administrator requires server and OS-level telemetry, including CPU, memory, and disk usage, network performance, and system logs.
Security engineer	The security engineer implements and manages security measures to help protect data and applications from threats. The security engineer handles everything from identity management to threat detection and response. They use telemetry about security events, including access logs, threat-detection alerts, vulnerability assessments, and compliance metrics.
Network administrator	The network administrator manages and maintains the cloud network to help ensure that data flows securely and efficiently between servers, applications, and users. The network administrator handles network configurations, monitors performance, and implements security measures. They require network-centric telemetry, including network traffic analysis, latency measurements, packet loss, and firewall logs.
Database administrator	The DBA manages and maintains databases to help ensure data integrity, performance, and availability. The DBA handles database backups and recovery

Role	Description
(DBA)	and optimizes queries for efficiency. They use telemetry about database performance and integrity, including query performance metrics, database response times, transaction logs, and backup or recovery status.
Developer	The developer designs, writes, tests, and maintains the software that runs on cloud platforms. The developer creates features and fixes bugs to help ensure that the application remains secure and performs well. They require application-specific telemetry, including error rates, latency, response times, user behavior analytics, and feature usage metrics.

Azure facilitation

Azure has many services that support the different [types of monitoring](#) that you need in your cloud environment. Each service targets one or more [roles](#). Combine services to provide the features that you need for a comprehensive monitoring environment.

[\[+\] Expand table](#)

Service	Description	Type	Roles
Azure Monitor	Azure Monitor is at the center of the Azure monitoring ecosystem. It's a comprehensive monitoring solution that you can use to collect, analyze, and respond to monitoring data from your cloud and on-premises environments. Azure Monitor provides complete monitoring of your infrastructure, network, and applications. It also provides a data platform and core features, such as data analysis, visualization, and alerting for other services.	Infrastructure, database, compliance	Cloud architect, platform engineer, system administrator, DBA
Application Insights	Application Insights is a feature of Azure Monitor that provides APM monitoring for your cloud applications.	APM	Developer
Azure Network Watcher	Network Watcher provides monitoring and visualization capabilities for network resources in Azure. Use this service to monitor, diagnose, and view metrics. You can also enable or disable logs for resources in an Azure virtual network.	Network	Network administrator
Microsoft Sentinel	Microsoft Sentinel is a cloud-native security information event management (SIEM) and security orchestration automated response (SOAR) solution. It ingests security telemetry	Security	Security engineer

Service	Description	Type	Roles
	from your Azure resources and other components to provide cyber-threat detection, investigation, response, and proactive hunting.		
Microsoft Defender XDR	Defender XDR includes Microsoft security solutions that are native to the Azure platform, client and server Microsoft operating systems, and applications including Office 365, Exchange Online, and SharePoint in Microsoft 365. Each security solution uses AI and machine learning to correlate telemetry and determine if investigations are necessary. When they detect unacceptable behavior, they take action to prevent disruption.	Security	Security engineer
Microsoft Cost Management	Cost Management is a suite of tools that you can use to analyze, monitor, and optimize your Microsoft Cloud costs. Cost Management is available to anyone that has access to a billing account, subscription, resource group, or management group.	Cost	Cloud architect
Azure Service Health	Service Health provides a health status of the services that your Azure resources rely on. It can inform you of any service outages and provide a personalized view of the health of your Azure services and regions.	Infrastructure	Cloud provider

Feedback

Was this page helpful?

 Yes

 No

Centralize management operations

Article • 10/06/2023

For most organizations, using a single Microsoft Entra tenant for all users simplifies management operations and reduces maintenance costs. This is because all management tasks can be by designated users, user groups, or service principals within that tenant.

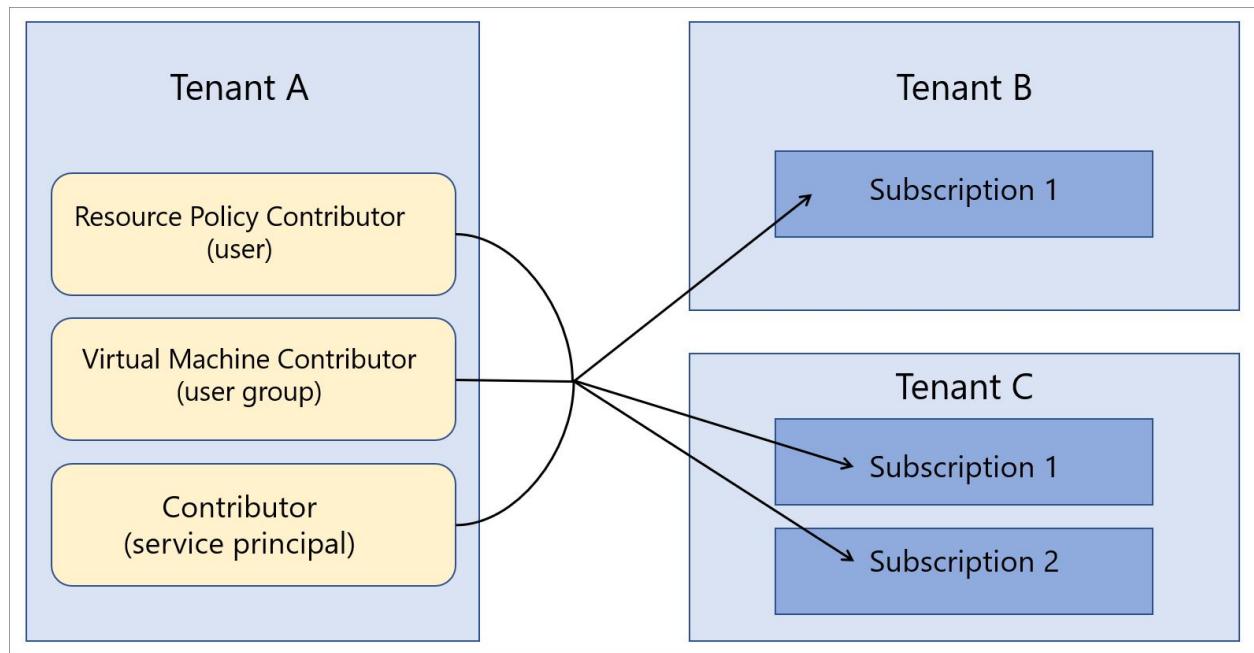
We recommend that you use only one Microsoft Entra tenant for your organization, if possible. However, some situations might require an organization to maintain multiple Microsoft Entra tenants for the following reasons:

- They are wholly independent subsidiaries.
- They're operating independently in multiple geographies.
- Certain legal or compliance requirements apply.
- There are acquisitions of other organizations (sometimes temporary until a long-term tenant consolidation strategy is defined).

When a multiple-tenant architecture is required, [Azure Lighthouse](#) provides a way to centralize and streamline management operations. Subscriptions from multiple tenants can be onboarded for [Azure delegated resource management](#). This option allows specified users in the managing tenant to perform [cross-tenant management functions](#) in a centralized and scalable manner.

For example, let's say your organization has a single tenant, `Tenant A`. The organization then acquires two additional tenants, `Tenant B` and `Tenant C`, and you have business reasons that require you to maintain them as separate tenants.

Your organization wants to use the same policy definitions, backup practices, and security processes across all tenants. Because you already have users (including user groups and service principals) that are responsible for performing these tasks within `Tenant A`, you can onboard all of the subscriptions within `Tenant B` and `Tenant C` so that those same users in `Tenant A` can perform those tasks. `Tenant A` then becomes the managing tenant for `Tenant B` and `Tenant C`.



For more information, see [Azure Lighthouse in enterprise scenarios](#).

Establish operations management processes

Article • 12/01/2022

As your enterprise begins to operate workloads in Azure, the next step is to establish a process for *operational management and fitness*. This process enumerates, implements, and iteratively reviews and optimizes the operational state for these workloads.

A process for operational fitness review ensures that the entire portfolio of workloads meet business commitments to performance, reliability, and cost. This process aligns the efforts of [central IT](#), [cloud center of excellence](#), and workload teams to deliver operational excellence at scale.

Establish a core process for operational fitness review

Create a process for operational fitness review to fully understand the problems that result from running workloads in a production environment, and how to remediate and resolve those problems. This article outlines a high-level process for operational fitness review that your enterprise can use to achieve this goal.

Operational fitness at Microsoft

From the outset, many teams across Microsoft have been involved in the development of the Azure platform. It's difficult to ensure quality and consistency for a project of such size and complexity. You need a robust process to enumerate and implement fundamental nonfunctional requirements on a regular basis.

The processes that Microsoft follows form the basis for the processes outlined in this article.

Understand roles and operating models

Operations management is a broad discipline involving multiple roles across the company. Depending on the organizations operating model, those roles may operate in a matrixed environment with a number of handoffs between centralized and decentralized operations teams.

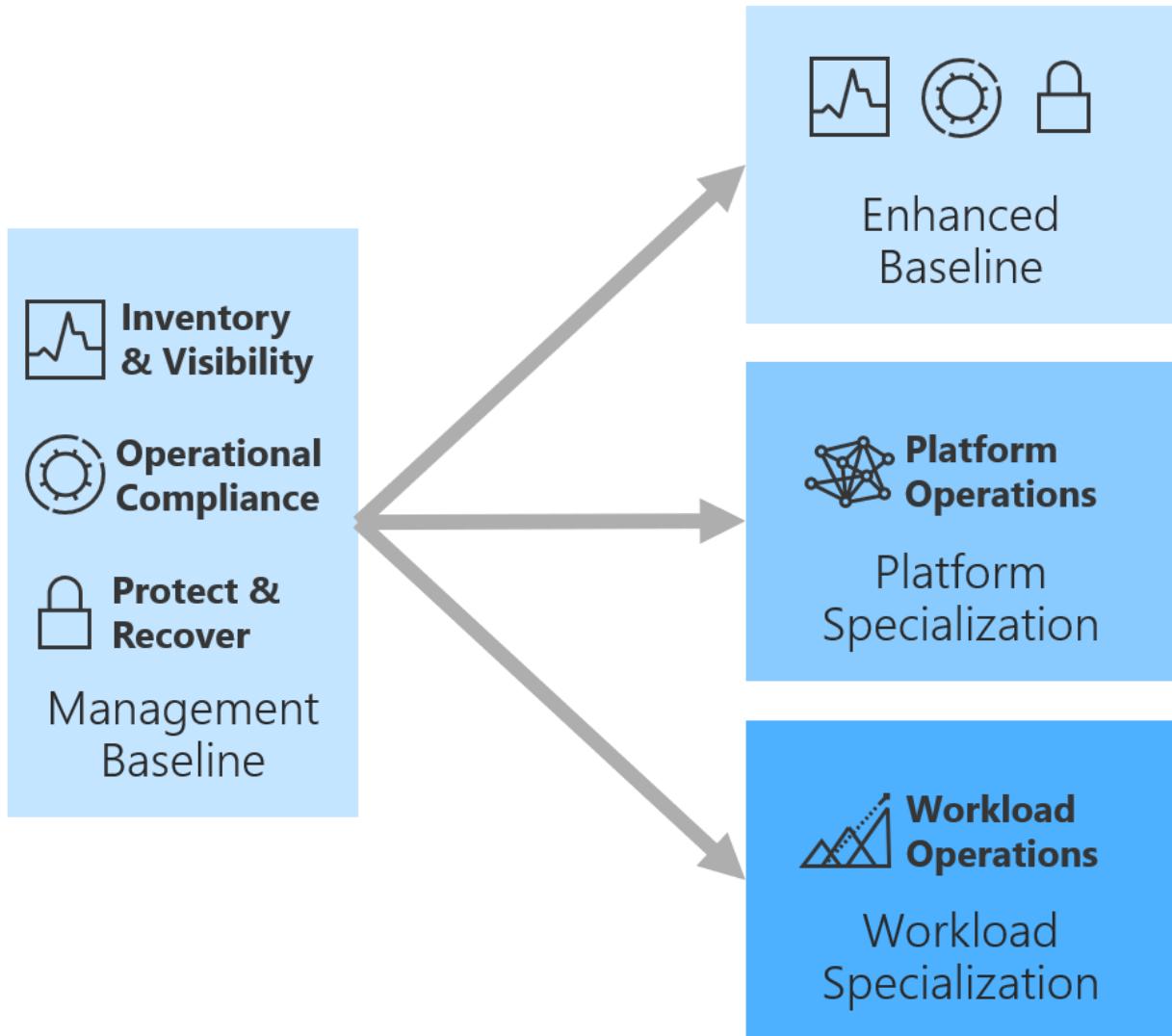
- **Central IT / CCoE:** This centralized technology function is responsible for the configuration, operations, governance, and security of all technology assets in the technology portfolio.
- **Cloud operations:** A function within the centralized technology organization, this operations function manages the health and operations of the technology portfolio. It's their responsibility to ensure the process runs smoothly, that each adjacent role in the process has the necessary tools, and that each of the subsequent roles is held accountable for expectations of this process.
- **Cloud strategy:** Provides knowledge of the business to identify and prioritize commitments to maintain operational requirements of various workloads. This role also compares the mitigation cost to the business impact, and drives the final decision on remediation.
- **Workload team:** Accountable for development and operations of discreet workloads which map to specific supporting applications, services, and infrastructure, whether on-premises or in the cloud. The role requires deep knowledge of the workload architecture.

Each organization's operating model determines the accountability and day-to-day activities of the roles above:

- **Centralized operations:** Central IT maintains full accountability for operations. Workload owners may have input to operations and configuration, but they have no access to change production environments. Only central IT and cloud operations can deliver operational change to improve operational fitness.
- **Decentralized operations:** Workload teams are fully accountable for operations, generally through a mature CI/CD pipeline and DevOps automation. In this model, there is no central support for configuration, operations, governance, or security. This approach to operations is out of scope for the Cloud Adoption Framework. This operating model should see the [Azure Well-Architected Framework](#) for operational guidance.
- **Enterprise operations:** The cloud center of excellence is accountable for operations. Cloud operations and workload teams each share responsibility for specific aspects of operational fitness.

Objective of the review

Operational fitness is evaluated across the portfolio using a few metrics: reliability, performance, and cost. Together, these properties allow for rapid evaluation of the health and fitness of all assets in the portfolio. These metrics are evaluated across the three elevations of operations management.



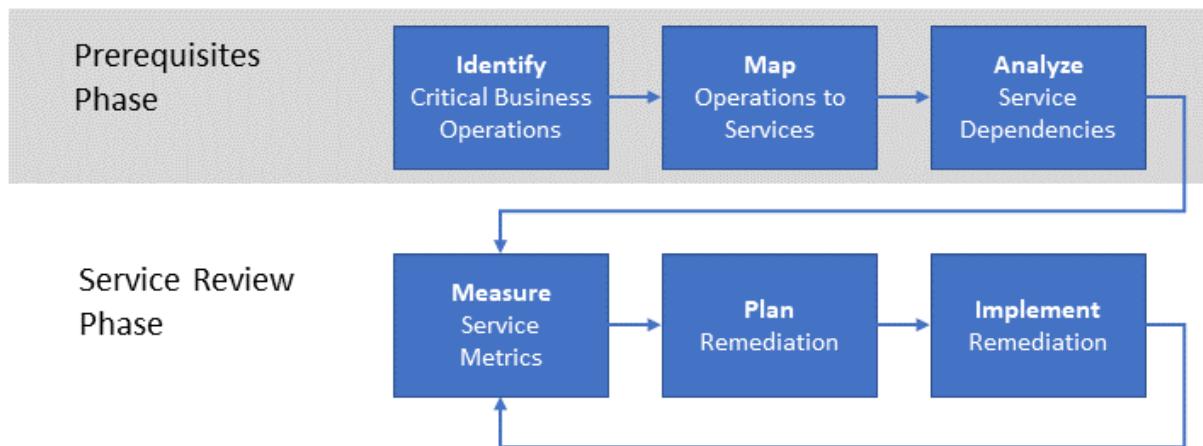
- **Operations baseline (or enhanced baseline)**: Evaluates operational fitness across all deployed assets regardless of their function. This broad view of operations allows for sweeping changes and big impacts, but is limited by a lack of visibility into the architecture of individual workloads. All resources deployed in the cloud should be covered by an operations baseline with regular support from cloud operations. Some environments may require a higher degree of operational support to meet the needs of the enhanced baseline.
- **Platform operations**: Evaluates operational fitness of centralized technology platforms. This view of operations is more refined since it considers the architecture of the platform and how changes to the solution will affect operational fitness. Changes to central technology platforms can have a broad downstream impact on supported workloads. All mission-critical platforms should receive dedicated support from a central IT team.
- **Workload operations**: Evaluates operational fitness of an individual workload. This view of operations is most refined and should be considered when operational fitness improvements require changes to the architecture of a workload. Workload operations should adhere to the principles of the [Azure Well-Architected](#)

[Framework](#). All mission-critical workloads with an active DevOps cycle should receive dedicated support from a workload team.

The objective of the operational fitness review is to regularly evaluate operational fitness at all levels. Identified improvements can then be applied at the appropriate level to inform the changes required to manage the overall portfolio.

Process for operational fitness review

The key to maintaining the performance and continuity of an enterprise's portfolio is to implement a process for operational fitness review.



At a high level, the process has two phases. In the *prerequisites phase*, the requirements are established and mapped to supporting services. This phase occurs infrequently: perhaps annually or when new operations are introduced. The output of the prerequisites phase is used in the *flow phase*. The flow phase occurs more frequently, such as monthly.

Prerequisites phase

The steps in this phase capture the requirements for conducting a regular review of the portfolio and any mission critical workloads.

1. **Identify critical business operations.** Identify the enterprise's mission-critical business operations based on agreed upon business commitments. Business operations are independent from any supporting service functionality. In other words, business operations represent the actual activities that the business needs to perform and that are supported by a set of IT services.

The term *mission-critical* (or *business-critical*) reflects a severe impact on the business if the operation is impeded. For example, an online retailer might have a

business operation, such as "enable a customer to add an item to a shopping cart" or "process a credit card payment." If either of these operations fails, a customer can't complete the transaction and the enterprise fails to realize sales.

2. **Map operations to services.** Map the critical business operations to the IT services (baseline, platform, or workload operations) that support them. Any technology platform or workload required to support a critical business function should also be identified to map operations and services to responsible teams.
3. **Analyze service dependencies.** Most business operations require orchestration among multiple supporting workloads and technology platforms. It's important to understand the dependencies between each set of supporting assets, and the flow of mission-critical transactions through these services.

Also consider the dependencies between on-premises services and Azure services. In the shopping-cart example, the inventory stock-management service might be hosted on-premises and ingest data entered by employees from a physical warehouse. However, it might store data off-premises in an Azure service, such as [Azure Storage](#), or a database, such as [Azure Cosmos DB](#).

An output from these activities is a set of *scorecard metrics* for operations management. The scorecard measures criteria such as reliability, performance, and cost. Scorecard metrics express the operational criteria that you expect the service to meet.

The scorecard should be expressed in simple terms to facilitate meaningful discussion between the business owners, cloud operations, and workload teams. For example, a scorecard metric for reliability might be color-coded based on attainment of the agreed SLA. Green means meeting the defined SLA, yellow means failing to meet the defined criteria but actively implementing a planned remediation, and red means failing to meet the defined criteria with no plan or action.

It's important to emphasize that these metrics should directly reflect business commitments.

Service-review phase

The service-review phase is the core of the operational fitness review. It involves these steps:

1. **Measure service metrics.** Use the scorecard metrics to monitor performance at each level of operations management, to ensure that the services meet the business commitments. Inventory and visibility services within the operations baseline is essential. If you can't monitor a set of resources with respect to the

business commitments, consider the corresponding scorecard metrics to be red. In this case, the first step for remediation is to implement the appropriate service monitoring. For example, if the business expects a service to operate with 99.99 percent availability, but there is no production telemetry in place to measure availability, assume that you're not meeting the requirement.

2. **Plan remediation.** For each business commitment for which metrics fall below an acceptable threshold, determine the appropriate operations team to complete the required remediation. That team is responsible for calculating the cost of remediating the service to bring operations to an acceptable level. If the cost of remediating the issue is greater than the budget allocated to that service, central IT / CCoE should review with the cloud strategy team to evaluate addition investments.
3. **Implement remediation.** After the cloud operations or workload team gain acceptance on a plan for remediation, implement it. Report the status of the implementation whenever you review scorecard metrics.

This process is iterative. The central IT / CCoE team is accountable for managing the process and reporting on progress to the cloud strategy team. This team should meet regularly to review existing remediation projects, kick off the fundamental review of new workloads, and track the enterprise's overall scorecard. The team should also have the authority to hold remediation teams (cloud operations or workload operations) accountable if they're behind schedule or fail to meet metrics.

Review meeting

We recommend that your operational fitness be reviewed on a regular basis. Central IT / CCoE and cloud operations team are required attendance in the review. Cloud strategy and workload operations teams are encouraged to attend but are operational. Example cadence, the core team might meet monthly to align on plans and hold various operations teams accountable. Quarterly, the cloud strategy and all workload teams could join to understand status and metrics.

Adapt the details of the process and meeting to fit your specific needs. We recommend the following considerations as a starting point:

- **Centralized operations:** Workload teams are unlikely to participate actively in the process, but should be included in any reports for visibility.
- **Decentralized operations:** Cloud operations team should share best practices used to improve operations of tech platforms with the workload teams. Workload teams

should share changes to their respective workloads to identify improvements that could be applied to tech platforms and the operations baseline.

Recommended resources

- [Azure Automanage](#). Azure Automanage automatically monitors operational fitness across the operations baseline and automates the application of various remediation strategies across the portfolio.
- [Azure Advisor](#). Azure Advisor provides personalized recommendations based on your usage and configurations to help optimize your resources. By default this tool provides recommendations across a subscription to improve the operations baseline. It can also be used more granularly to identify improvements to tech platforms or individual workloads.
- [Microsoft Azure Well-Architected Framework](#): Guidance to improve workload operations or to guide decentralized operations.

Resiliency checklist for specific Azure services

Article • 07/26/2023

Resiliency is the ability of a system to recover from failures and continue to function. Every technology has its own particular failure modes, which you must consider when designing and implementing your application. Use this checklist to review the resiliency considerations for specific Azure services. For more information about designing resilient applications, see [Design reliable Azure applications](#).

App Service

Use Standard or Premium tier. These tiers support staging slots and automated backups. For more information, see [Azure App Service plans in-depth overview](#)

Avoid scaling up or down. Instead, select a tier and instance size that meet your performance requirements under typical load, and then [scale out](#) the instances to handle changes in traffic volume. Scaling up and down may trigger an application restart.

Store configuration as app settings. Use app settings to hold configuration settings as app settings. Define the settings in your Resource Manager templates, or using PowerShell, so that you can apply them as part of an automated deployment / update process, which is more reliable. For more information, see [Configure web apps in Azure App Service](#).

Create separate App Service plans for production and test. Don't use slots on your production deployment for testing. All apps within the same App Service plan share the same VM instances. If you put production and test deployments in the same plan, it can negatively affect the production deployment. For example, load tests might degrade the live production site. By putting test deployments into a separate plan, you isolate them from the production version.

Separate web apps from web APIs. If your solution has both a web front end and a web API, consider decomposing them into separate App Service apps. This design makes it easier to decompose the solution by workload. You can run the web app and the API in separate App Service plans, so they can be scaled independently. If you don't need that level of scalability at first, you can deploy the apps into the same plan, and move them into separate plans later, if needed.

Deploy zone-redundant App Service plans. In supported regions, App Service plans can be deployed as zone redundant, which means that the instances are automatically distributed across availability zones. App Service automatically distributes traffic between the zones, and handles failover if a zone experiences an outage. For more information, see [Migrate App Service to availability zone support](#).

Avoid using the App Service backup feature to back up Azure SQL databases. Instead, use [SQL Database automated backups](#). App Service backup exports the database to a SQL BACPAC file, which costs DTUs.

Deploy to a staging slot. Create a deployment slot for staging. Deploy application updates to the staging slot, and verify the deployment before swapping it into production. This reduces the chance of a bad update in production. It also ensures that all instances are warmed up before being swapped into production. Many applications have a significant warmup and cold-start time. For more information, see [Set up staging environments for web apps in Azure App Service](#).

Create a deployment slot to hold the last-known-good (LKG) deployment. When you deploy an update to production, move the previous production deployment into the LKG slot. This makes it easier to roll back a bad deployment. If you discover a problem later, you can quickly revert to the LKG version. For more information, see [Basic web application](#).

Enable diagnostics logging, including application logging and web server logging. Logging is important for monitoring and diagnostics. See [Enable diagnostics logging for web apps in Azure App Service](#)

Log to blob storage. This makes it easier to collect and analyze the data.

Create a separate storage account for logs. Don't use the same storage account for logs and application data. This helps to prevent logging from reducing application performance.

Monitor performance. Use a performance monitoring service such as [New Relic](#) or [Application Insights](#) to monitor application performance and behavior under load. Performance monitoring gives you real-time insight into the application. It enables you to diagnose issues and perform root-cause analysis of failures.

Azure Load Balancer

Select Standard SKU. Standard Load Balancer provides a dimension of reliability that Basic does not - that of availability zones and zone resiliency. This means when a zone goes down, your zone-redundant Standard Load Balancer will not be impacted. This

ensures your deployments can withstand zone failures within a region. In addition, Standard Load Balancer supports global load balancing ensuring your application is not impacted by region failures either.

Provision at least two instances. Deploy Azure LB with at least two instances in the backend. A single instance could result in a single point of failure. In order to build for scale, you might want to pair LB with Virtual Machine Scale Sets.

Use outbound rules. Outbound rules ensure that you are not faced with connection failures as a result of SNAT port exhaustion. [Learn more about outbound connectivity](#). While outbound rules will help improve the solution for small to mid size deployments, for production workloads, we recommend coupling Standard Load Balancer or any subnet deployment with [VNet NAT](#).

Application Gateway

Provision at least two instances. Deploy Application Gateway with at least two instances. A single instance is a single point of failure. Use two or more instances for redundancy and scalability. In order to qualify for the [SLA](#), you must provision two or more medium or larger instances.

Azure Cosmos DB

Configure zone redundancy. When you use zone redundancy, Azure Cosmos DB synchronously replicates all writes across availability zones. It automatically fails over in the event of a zone outage. For more information, see [Achieve high availability with Azure Cosmos DB](#).

Replicate the database across regions. Azure Cosmos DB allows you to associate any number of Azure regions with an Azure Cosmos DB database account. An Azure Cosmos DB database can have one write region and multiple read regions. If there is a failure in the write region, you can read from another replica. The Client SDK handles this automatically. You can also fail over the write region to another region. For more information, see [How to distribute data globally with Azure Cosmos DB](#).

Event Hubs

Use checkpoints. An event consumer should write its current position to persistent storage at some predefined interval. That way, if the consumer experiences a fault (for example, the consumer crashes, or the host fails), then a new instance can resume

reading the stream from the last recorded position. For more information, see [Event consumers](#).

Handle duplicate messages. If an event consumer fails, message processing is resumed from the last recorded checkpoint. Any messages that were already processed after the last checkpoint will be processed again. Therefore, your message processing logic must be idempotent, or the application must be able to deduplicate messages.

Handle exceptions.. An event consumer typically processes a batch of messages in a loop. You should handle exceptions within this processing loop to avoid losing an entire batch of messages if a single message causes an exception.

Use a dead-letter queue. If processing a message results in a nontransient failure, put the message onto a dead-letter queue, so that you can track the status. Depending on the scenario, you might retry the message later, apply a compensating transaction, or take some other action. Note that Event Hubs does not have any built-in dead-letter queue functionality. You can use Azure Queue Storage or Service Bus to implement a dead-letter queue, or use Azure Functions or some other eventing mechanism.

Configure zone redundancy. When zone redundancy is enabled on your namespace, Event Hubs automatically replicates changes between multiple availability zones. If one availability zone fails, failover happens automatically. For more information, see [Availability zones](#).

Implement disaster recovery by failing over to a secondary Event Hubs namespace. For more information, see [Azure Event Hubs Geo-disaster recovery](#).

Azure Cache for Redis

Configure zone redundancy. When zone redundancy is enabled on your cache, Azure Cache for Redis spreads the virtual machines that host your cache across multiple availability zones. Zone redundancy provides high availability and fault tolerance in the event of a data center outage. For more information, see [Enable zone redundancy for Azure Cache for Redis](#).

Configure Geo-replication. Geo-replication provides a mechanism for linking two Premium-tier Azure Cache for Redis instances. Data written to the primary cache is replicated to a secondary read-only cache. For more information, see [How to configure geo-replication for Azure Cache for Redis](#)

Configure data persistence. Redis persistence allows you to persist data stored in Redis. You can also take snapshots and back up the data, which you can load in case of a

hardware failure. For more information, see [How to configure data persistence for a Premium-tier Azure Cache for Redis](#)

If you are using Azure Cache for Redis as a temporary data cache and not as a persistent store, these recommendations may not apply.

Cognitive Search

Provision more than one replica. Use at least two replicas for read high-availability, or three for read-write high-availability.

Use zone redundancy. You can deploy Cognitive Search replicas across multiple availability zones. This approach helps your service to remain operational even when data center outages occur. For more information, see [Reliability in Azure Cognitive Search](#)

Configure indexers for multi-region deployments. If you have a multi-region deployment, consider your options for continuity in indexing.

- If the data source is geo-replicated, you should generally point each indexer of each regional Azure Cognitive Search service to its local data source replica. However, that approach is not recommended for large datasets stored in Azure SQL Database. The reason is that Azure Cognitive Search cannot perform incremental indexing from secondary SQL Database replicas, only from primary replicas. Instead, point all indexers to the primary replica. After a failover, point the Azure Cognitive Search indexers at the new primary replica.
- If the data source is not geo-replicated, point multiple indexers at the same data source, so that Azure Cognitive Search services in multiple regions continuously and independently index from the data source. For more information, see [Azure Search performance and optimization considerations](#).

Service Bus

Use Premium tier for production workloads. [Service Bus Premium Messaging](#) provides dedicated and reserved processing resources, and memory capacity to support predictable performance and throughput. Premium Messaging tier also gives you access to new features that are available only to premium customers at first. You can decide the number of messaging units based on expected workloads.

Handle duplicate messages. If a publisher fails immediately after sending a message, or experiences network or system issues, it may erroneously fail to record that the message

was delivered, and may send the same message to the system twice. Service Bus can handle this issue by enabling duplicate detection. For more information, see [Duplicate detection](#).

Handle exceptions. Messaging APIs generate exceptions when a user error, configuration error, or other error occurs. The client code (senders and receivers) should handle these exceptions in their code. This is especially important in batch processing, where exception handling can be used to avoid losing an entire batch of messages. For more information, see [Service Bus messaging exceptions](#).

Retry policy. Service Bus allows you to pick the best retry policy for your applications. The default policy is to allow 9 maximum retry attempts, and wait for 30 seconds but this can be further adjusted. For more information, see [Retry policy – Service Bus](#).

Use a dead-letter queue. If a message cannot be processed or delivered to any receiver after multiple retries, it is moved to a dead letter queue. Implement a process to read messages from the dead letter queue, inspect them, and remediate the problem. Depending on the scenario, you might retry the message as-is, make changes and retry, or discard the message. For more information, see [Overview of Service Bus dead-letter queues](#).

Use zone redundancy. When zone redundancy is enabled on your namespace, Service Bus automatically replicates changes between multiple availability zones. If one availability zone fails, failover happens automatically. For more information, see [Best practices for insulating applications against Service Bus outages and disasters](#).

Use Geo-Disaster Recovery. Geo-disaster recovery ensures that data processing continues to operate in a different region or datacenter if an entire Azure region or datacenter becomes unavailable due to a disaster. For more information, see [Azure Service Bus Geo-disaster recovery](#).

Storage

Use zone-redundant storage. Zone-redundant storage (ZRS) copies your data synchronously across three Azure availability zones in the primary region. If an availability zone experiences an outage, Azure Storage automatically fails over to an alternative zone. For more information, see [Azure Storage redundancy](#).

When using geo-redundancy, configure read-access. If you use a multi-region architecture, use a suitable storage tier for global redundancy. With RA-GRS or RA-GZRS, your data is replicated to a secondary region. RA-GRS uses locally redundant storage (LRS) in the primary region, while RA-GZRS uses zone-redundant storage (ZRS)

in the primary region. Both configurations provide read-only access to your data in the secondary region. If there is a storage outage in the primary region, the application can read the data from the secondary region if you have designed it for this possibility. For more information, see [Azure Storage redundancy](#).

For VM disks, use managed disks. [Managed disks](#) provide better reliability for VMs in an availability set, because the disks are sufficiently isolated from each other to avoid single points of failure. Also, managed disks aren't subject to the IOPS limits of VHDs created in a storage account. For more information, see [Manage the availability of Windows virtual machines in Azure](#).

For Queue storage, create a backup queue in another region. For Queue Storage, a read-only replica has limited use, because you can't queue or dequeue items. Instead, create a backup queue in a storage account in another region. If there is an Azure Storage outage, the application can use the backup queue, until the primary region becomes available again. That way, the application can continue to process new requests during the outage.

SQL Database

Use Standard or Premium tier. These tiers provide a longer point-in-time restore period (35 days). For more information, see [SQL Database options and performance](#).

Enable SQL Database auditing. Auditing can be used to diagnose malicious attacks or human error. For more information, see [Get started with SQL database auditing](#).

Use Active Geo-Replication Use Active Geo-Replication to create a readable secondary in a different region. If your primary database fails, or simply needs to be taken offline, perform a manual failover to the secondary database. Until you fail over, the secondary database remains read-only. For more information, see [SQL Database Active Geo-Replication](#).

Use sharding. Consider using sharding to partition the database horizontally. Sharding can provide fault isolation. For more information, see [Scaling out with Azure SQL Database](#).

Use point-in-time restore to recover from human error. Point-in-time restore returns your database to an earlier point in time. For more information, see [Recover an Azure SQL database using automated database backups](#).

Use geo-restore to recover from a service outage. Geo-restore restores a database from a geo-redundant backup. For more information, see [Recover an Azure SQL database using automated database backups](#).

Azure Synapse Analytics

Do not disable geo-backup. By default, Synapse Analytics takes a full backup of your data every 24 hours for disaster recovery. It is not recommended to turn this feature off. For more information, see [Geo-backups](#).

SQL Server running in a VM

Replicate the database. Use SQL Server Always On availability groups to replicate the database. Provides high availability if one SQL Server instance fails. For more information, see [Run Windows VMs for an N-tier application](#)

Back up the database. If you are already using [Azure Backup](#) to back up your VMs, consider using [Azure Backup for SQL Server workloads using DPM](#). With this approach, there is one backup administrator role for the organization and a unified recovery procedure for VMs and SQL Server. Otherwise, use [SQL Server Managed Backup to Microsoft Azure](#).

Traffic Manager

Perform manual failback. After a Traffic Manager failover, perform manual failback, rather than automatically failing back. Before failing back, verify that all application subsystems are healthy. Otherwise, you can create a situation where the application flips back and forth between datacenters. For more information, see [Run VMs in multiple regions for high availability](#).

Create a health probe endpoint. Create a custom endpoint that reports on the overall health of the application. This enables Traffic Manager to fail over if any critical path fails, not just the front end. The endpoint should return an HTTP error code if any critical dependency is unhealthy or unreachable. Don't report errors for non-critical services, however. Otherwise, the health probe might trigger failover when it's not needed, creating false positives. For more information, see [Traffic Manager endpoint monitoring and failover](#).

Virtual Machines

Avoid running a production workload on a single VM. A single VM deployment is not resilient to planned or unplanned maintenance. Instead, put multiple VMs in an availability set or [virtual machine scale set](#), with a load balancer in front.

Specify an availability set when you provision the VM. Currently, there is no way to add a VM to an availability set after the VM is provisioned. When you add a new VM to an existing availability set, make sure to create a NIC for the VM, and add the NIC to the back-end address pool on the load balancer. Otherwise, the load balancer won't route network traffic to that VM.

Put each application tier into a separate Availability Set. In an N-tier application, don't put VMs from different tiers into the same availability set. VMs in an availability set are placed across fault domains (FDs) and update domains (UD). However, to get the redundancy benefit of FDs and UD, every VM in the availability set must be able to handle the same client requests.

Replicate VMs using Azure Site Recovery. When you replicate Azure VMs using [Site Recovery](#), all the VM disks are continuously replicated to the target region asynchronously. The recovery points are created every few minutes. This gives you a Recovery Point Objective (RPO) in the order of minutes. You can conduct disaster recovery drills as many times as you want, without affecting the production application or the ongoing replication. For more information, see [Run a disaster recovery drill to Azure](#).

Choose the right VM size based on performance requirements. When moving an existing workload to Azure, start with the VM size that's the closest match to your on-premises servers. Then measure the performance of your actual workload with respect to CPU, memory, and disk IOPS, and adjust the size if needed. This helps to ensure the application behaves as expected in a cloud environment. Also, if you need multiple NICs, be aware of the NIC limit for each size.

Use managed disks for VHDs. [Managed disks](#) provide better reliability for VMs in an availability set, because the disks are sufficiently isolated from each other to avoid single points of failure. Also, managed disks aren't subject to the IOPS limits of VHDs created in a storage account. For more information, see [Manage the availability of Windows virtual machines in Azure](#).

Install applications on a data disk, not the OS disk. Otherwise, you may reach the disk size limit.

Use Azure Backup to back up VMs. Backups protect against accidental data loss. For more information, see [Protect Azure VMs with a Recovery Services vault](#).

Enable diagnostic logs. Include basic health metrics, infrastructure logs, and [boot diagnostics](#). Boot diagnostics can help you diagnose a boot failure if your VM gets into a nonbootable state. For more information, see [Overview of Azure Diagnostic Logs](#).

Configure Azure Monitor. Collect and analyze monitoring data from Azure virtual machines including the guest operating system and the workloads that run in it, see [Azure Monitor](#) and [Quickstart: Azure Monitor](#).

Virtual Network

To allow or block public IP addresses, add a network security group to the subnet. Block access from malicious users, or allow access only from users who have privilege to access the application.

Create a custom health probe. Load Balancer Health Probes can test either HTTP or TCP. If a VM runs an HTTP server, the HTTP probe is a better indicator of health status than a TCP probe. For an HTTP probe, use a custom endpoint that reports the overall health of the application, including all critical dependencies. For more information, see [Azure Load Balancer overview](#).

Don't block the health probe. The Load Balancer Health probe is sent from a known IP address, 168.63.129.16. Don't block traffic to or from this IP in any firewall policies or network security group rules. Blocking the health probe would cause the load balancer to remove the VM from rotation.

Enable Load Balancer logging. The logs show how many VMs on the back-end are not receiving network traffic due to failed probe responses. For more information, see [Log analytics for Azure Load Balancer](#).

Failure mode analysis for Azure applications

Article • 07/26/2023

Failure mode analysis (FMA) is a process for building resiliency into a system, by identifying possible failure points in the system. The FMA should be part of the architecture and design phases, so that you can build failure recovery into the system from the beginning.

Here is the general process to conduct an FMA:

1. Identify all of the components in the system. Include external dependencies, such as identity providers, third-party services, and so on.
2. For each component, identify potential failures that could occur. A single component may have more than one failure mode. For example, you should consider read failures and write failures separately, because the impact and possible mitigation steps will be different.
3. Rate each failure mode according to its overall risk. Consider these factors:
 - What is the likelihood of the failure. Is it relatively common? Extremely rare? You don't need exact numbers; the purpose is to help rank the priority.
 - What is the impact on the application, in terms of availability, data loss, monetary cost, and business disruption?
4. For each failure mode, determine how the application will respond and recover. Consider tradeoffs in cost and application complexity.

As a starting point for your FMA process, this article contains a catalog of potential failure modes and their mitigation steps. The catalog is organized by technology or Azure service, plus a general category for application-level design. The catalog is not exhaustive, but covers many of the core Azure services.

App Service

App Service app shuts down.

Detection. Possible causes:

- Expected shutdown

- An operator shuts down the application; for example, using the Azure portal.
- The app was unloaded because it was idle. (Only if the `Always On` setting is disabled.)
- Unexpected shutdown
 - The app crashes.
 - An App Service VM instance becomes unavailable.

`Application_End` logging will catch the app domain shutdown (soft process crash) and is the only way to catch the application domain shutdowns.

Recovery:

- If the shutdown was expected, use the application's shutdown event to shut down gracefully. For example, in ASP.NET, use the `Application_End` method.
- If the application was unloaded while idle, it is automatically restarted on the next request. However, you will incur the "cold start" cost.
- To prevent the application from being unloaded while idle, enable the `Always On` setting in the web app. See [Configure web apps in Azure App Service](#).
- To prevent an operator from shutting down the app, set a resource lock with `ReadOnly` level. See [Lock resources with Azure Resource Manager](#).
- If the app crashes or an App Service VM becomes unavailable, App Service automatically restarts the app.

Diagnostics. Application logs and web server logs. See [Enable diagnostics logging for web apps in Azure App Service](#).

A particular user repeatedly makes bad requests or overloads the system.

Detection. Authenticate users and include user ID in application logs.

Recovery:

- Use [Azure API Management](#) to throttle requests from the user. See [Advanced request throttling with Azure API Management](#)
- Block the user.

Diagnostics. Log all authentication requests.

A bad update was deployed.

Detection. Monitor the application health through the Azure portal (see [Monitor Azure web app performance](#)) or implement the [health endpoint monitoring pattern](#).

Recovery: Use multiple [deployment slots](#) and roll back to the last-known-good deployment. For more information, see [Basic web application](#).

Azure Active Directory

OpenID Connect authentication fails.

Detection. Possible failure modes include:

1. Azure AD is not available, or cannot be reached due to a network problem.
Redirection to the authentication endpoint fails, and the OpenID Connect middleware throws an exception.
2. Azure AD tenant does not exist. Redirection to the authentication endpoint returns an HTTP error code, and the OpenID Connect middleware throws an exception.
3. User cannot authenticate. No detection strategy is necessary; Azure AD handles login failures.

Recovery:

1. Catch unhandled exceptions from the middleware.
2. Handle `AuthenticationFailed` events.
3. Redirect the user to an error page.
4. User retries.

Azure Search

Writing data to Azure Search fails.

Detection. Catch `Microsoft.Rest.Azure.CloudException` errors.

Recovery:

The [Search .NET SDK](#) automatically retries after transient failures. Any exceptions thrown by the client SDK should be treated as nontransient errors.

The default retry policy uses exponential back-off. To use a different retry policy, call `SetRetryPolicy` on the `SearchIndexClient` or `SearchServiceClient` class. For more information, see [Automatic Retries](#).

Diagnostics. Use [Search Traffic Analytics](#).

Reading data from Azure Search fails.

Detection. Catch `Microsoft.Rest.Azure.CloudException` errors.

Recovery:

The [Search .NET SDK](#) automatically retries after transient failures. Any exceptions thrown by the client SDK should be treated as nontransient errors.

The default retry policy uses exponential back-off. To use a different retry policy, call `SetRetryPolicy` on the `SearchIndexClient` or `SearchServiceClient` class. For more information, see [Automatic Retries](#).

Diagnostics. Use [Search Traffic Analytics](#).

Cassandra

Reading or writing to a node fails.

Detection. Catch the exception. For .NET clients, this will typically be `System.Web.HttpException`. Other client may have other exception types. For more information, see [Cassandra error handling done right](#).

Recovery:

- Each [Cassandra client](#) has its own retry policies and capabilities. For more information, see [Cassandra error handling done right](#).
- Use a rack-aware deployment, with data nodes distributed across the fault domains.
- Deploy to multiple regions with local quorum consistency. If a nontransient failure occurs, fail over to another region.

Diagnostics. Application logs

Cloud Service

Web or worker roles are unexpectedly being shut down.

Detection. The `RoleEnvironmentStopping` event is fired.

Recovery. Override the `RoleEntryPoint.OnStop` method to gracefully clean up. For more information, see [The Right Way to Handle Azure OnStop Events](#) (blog).

Azure Cosmos DB

Reading data fails.

Detection. Catch `System.Net.Http.HttpRequestException` or `Microsoft.Azure.Documents.DocumentClientException`.

Recovery:

- The SDK automatically retries failed attempts. To set the number of retries and the maximum wait time, configure `ConnectionPolicy.RetryOptions`. Exceptions that the client raises are either beyond the retry policy or are not transient errors.
- If Azure Cosmos DB throttles the client, it returns an HTTP 429 error. Check the status code in the `DocumentClientException`. If you are getting error 429 consistently, consider increasing the throughput value of the collection.
 - If you are using the MongoDB API, the service returns error code 16500 when throttling.
- Enable zone redundancy when you work with a region that supports availability zones. When you use zone redundancy, Azure Cosmos DB automatically fails over in the event of a zone outage. For more information, see [Achieve high availability with Azure Cosmos DB](#).
- If you're designing a multi-region solution, replicate the Azure Cosmos DB database across two or more regions. All replicas are readable. Using the client SDKs, specify the `PreferredLocations` parameter. This is an ordered list of Azure regions. All reads will be sent to the first available region in the list. If the request fails, the client will try the other regions in the list, in order. For more information, see [How to set up global distribution in Azure Cosmos DB for NoSQL](#).

Diagnostics. Log all errors on the client side.

Writing data fails.

Detection. Catch `System.Net.Http.HttpRequestException` or `Microsoft.Azure.Documents.DocumentClientException`.

Recovery:

- The SDK automatically retries failed attempts. To set the number of retries and the maximum wait time, configure `ConnectionPolicy.RetryOptions`. Exceptions that the client raises are either beyond the retry policy or are not transient errors.
- If Azure Cosmos DB throttles the client, it returns an HTTP 429 error. Check the status code in the `DocumentClientException`. If you are getting error 429 consistently, consider increasing the throughput value of the collection.
- Enable zone redundancy when you work with a region that supports availability zones. When you use zone redundancy, Azure Cosmos DB synchronously replicates all writes across availability zones. For more information, see [Achieve high availability with Azure Cosmos DB](#).
- If you're designing a multi-region solution, replicate the Azure Cosmos DB database across two or more regions. If the primary region fails, another region will be promoted to write. You can also trigger a failover manually. The SDK does automatic discovery and routing, so application code continues to work after a failover. During the failover period (typically minutes), write operations will have higher latency, as the SDK finds the new write region. For more information, see [How to set up global distribution in Azure Cosmos DB for NoSQL](#).
- As a fallback, persist the document to a backup queue, and process the queue later.

Diagnostics. Log all errors on the client side.

Queue storage

Writing a message to Azure Queue storage fails consistently.

Detection. After N retry attempts, the write operation still fails.

Recovery:

- Store the data in a local cache, and forward the writes to storage later, when the service becomes available.
- Create a secondary queue, and write to that queue if the primary queue is unavailable.

Diagnostics. Use [storage metrics](#).

The application cannot process a particular message from the queue.

Detection. Application specific. For example, the message contains invalid data, or the business logic fails for some reason.

Recovery:

Move the message to a separate queue. Run a separate process to examine the messages in that queue.

Consider using Azure Service Bus Messaging queues, which provides a [dead-letter queue](#) functionality for this purpose.

 **Note**

If you are using Storage queues with WebJobs, the WebJobs SDK provides built-in poison message handling. See [How to use Azure queue storage with the WebJobs SDK](#).

Diagnostics. Use application logging.

Azure Cache for Redis

Reading from the cache fails.

Detection. Catch `StackExchange.Redis.RedisConnectionException`.

Recovery:

1. Retry on transient failures. Azure Cache for Redis supports built-in retry. For more information, see [Azure Cache for Redis retry guidelines](#).
2. Treat nontransient failures as a cache miss, and fall back to the original data source.

Diagnostics. Use [Azure Cache for Redis diagnostics](#).

Writing to the cache fails.

Detection. Catch `StackExchange.Redis.RedisConnectionException`.

Recovery:

1. Retry on transient failures. Azure Cache for Redis supports built-in retry. For more information, see [Azure Cache for Redis retry guidelines](#).

2. If the error is nontransient, ignore it and let other transactions write to the cache later.

Diagnostics. Use [Azure Cache for Redis diagnostics](#).

SQL Database

Cannot connect to the database in the primary region.

Detection. Connection fails.

Recovery:

- **Enable zone redundancy.** By enabling zone redundancy, Azure SQL Database automatically replicates your writes across multiple Azure availability zones within supported regions. For more information, see [Zone-redundant availability](#).
- **Enable geo-replication.** If you're designing a multi-region solution, consider enabling SQL Database active geo-replication.

Prerequisite: The database must be configured for active geo-replication. See [SQL Database Active Geo-Replication](#).

- For queries, read from a secondary replica.
- For inserts and updates, manually fail over to a secondary replica. See [Initiate a planned or unplanned failover for Azure SQL Database](#).

The replica uses a different connection string, so you will need to update the connection string in your application.

Client runs out of connections in the connection pool.

Detection. Catch `System.InvalidOperationException` errors.

Recovery:

- Retry the operation.
- As a mitigation plan, isolate the connection pools for each use case, so that one use case can't dominate all the connections.
- Increase the maximum connection pools.

Diagnostics. Application logs.

Database connection limit is reached.

Detection. Azure SQL Database limits the number of concurrent workers, logins, and sessions. The limits depend on the service tier. For more information, see [Azure SQL Database resource limits](#).

To detect these errors, catch `System.Data.SqlClient.SqlException` and check the value of `SqlException.Number` for the SQL error code. For a list of relevant error codes, see [SQL error codes for SQL Database client applications: Database connection error and other issues](#).

Recovery. These errors are considered transient, so retrying may resolve the issue. If you consistently hit these errors, consider scaling the database.

Diagnostics. - The [sys.event_log](#) query returns successful database connections, connection failures, and deadlocks.

- Create an [alert rule](#) for failed connections.
- Enable [SQL Database auditing](#) and check for failed logins.

Service Bus Messaging

Reading a message from a Service Bus queue fails.

Detection. Catch exceptions from the client SDK. The base class for Service Bus exceptions is [MessagingException](#). If the error is transient, the `IsTransient` property is true.

For more information, see [Service Bus messaging exceptions](#).

Recovery:

1. Retry on transient failures. See [Service Bus retry guidelines](#).
2. Messages that cannot be delivered to any receiver are placed in a *dead-letter queue*. Use this queue to see which messages could not be received. There is no automatic cleanup of the dead-letter queue. Messages remain there until you explicitly retrieve them. See [Overview of Service Bus dead-letter queues](#).

Writing a message to a Service Bus queue fails.

Detection. Catch exceptions from the client SDK. The base class for Service Bus exceptions is [MessagingException](#). If the error is transient, the `IsTransient` property is true.

For more information, see [Service Bus messaging exceptions](#).

Recovery:

- The Service Bus client automatically retries after transient errors. By default, it uses exponential back-off. After the maximum retry count or maximum timeout period, the client throws an exception. For more information, see [Service Bus retry guidelines](#).
- If the queue quota is exceeded, the client throws [QuotaExceededException](#). The exception message gives more details. Drain some messages from the queue before retrying, and consider using the Circuit Breaker pattern to avoid continued retries while the quota is exceeded. Also, make sure the [BrokeredMessage.TimeToLive](#) property is not set too high.
- Within a region, resiliency can be improved by using [partitioned queues or topics](#). A non-partitioned queue or topic is assigned to one messaging store. If this messaging store is unavailable, all operations on that queue or topic will fail. A partitioned queue or topic is partitioned across multiple messaging stores.
- Use zone redundancy to automatically replicate changes between multiple availability zones. If one availability zone fails, failover happens automatically. For more information, see [Best practices for insulating applications against Service Bus outages and disasters](#).
 - Active replication: The client sends every message to both queues. The receiver listens on both queues. Tag messages with a unique identifier, so the client can discard duplicate messages.
 - Passive replication: The client sends the message to one queue. If there is an error, the client falls back to the other queue. The receiver listens on both queues. This approach reduces the number of duplicate messages that are sent. However, the receiver must still handle duplicate messages.
- If you're designing a multi-region solution, create two Service Bus namespaces in different regions, and replicate the messages. You can use either active replication or passive replication.
 - Active replication: The client sends every message to both queues. The receiver listens on both queues. Tag messages with a unique identifier, so the client can discard duplicate messages.
 - Passive replication: The client sends the message to one queue. If there is an error, the client falls back to the other queue. The receiver listens on both queues. This approach reduces the number of duplicate messages that are sent. However, the receiver must still handle duplicate messages.

For more information, see [GeoReplication sample](#) and [Best practices for insulating applications against Service Bus outages and disasters](#).

Duplicate message.

Detection. Examine the `MessageId` and `DeliveryCount` properties of the message.

Recovery:

- If possible, design your message processing operations to be idempotent. Otherwise, store message IDs of messages that are already processed, and check the ID before processing a message.
- Enable duplicate detection, by creating the queue with `RequiresDuplicateDetection` set to true. With this setting, Service Bus automatically deletes any message that is sent with the same `MessageId` as a previous message.
Note the following:
 - This setting prevents duplicate messages from being put into the queue. It doesn't prevent a receiver from processing the same message more than once.
 - Duplicate detection has a time window. If a duplicate is sent beyond this window, it won't be detected.

Diagnostics. Log duplicated messages.

The application can't process a particular message from the queue.

Detection. Application specific. For example, the message contains invalid data, or the business logic fails for some reason.

Recovery:

There are two failure modes to consider.

- The receiver detects the failure. In this case, move the message to the dead-letter queue. Later, run a separate process to examine the messages in the dead-letter queue.
- The receiver fails in the middle of processing the message — for example, due to an unhandled exception. To handle this case, use `PeekLock` mode. In this mode, if the lock expires, the message becomes available to other receivers. If the message exceeds the maximum delivery count or the time-to-live, the message is automatically moved to the dead-letter queue.

For more information, see [Overview of Service Bus dead-letter queues](#).

Diagnostics. Whenever the application moves a message to the dead-letter queue, write an event to the application logs.

Service Fabric

A request to a service fails.

Detection. The service returns an error.

Recovery:

- Locate a proxy again (`ServiceProxy` or `ActorProxy`) and call the service/actor method again.
- **Stateful service.** Wrap operations on reliable collections in a transaction. If there is an error, the transaction will be rolled back. The request, if pulled from a queue, will be processed again.
- **Stateless service.** If the service persists data to an external store, all operations need to be idempotent.

Diagnostics. Application log

Service Fabric node is shut down.

Detection. A cancellation token is passed to the service's `RunAsync` method. Service Fabric cancels the task before shutting down the node.

Recovery. Use the cancellation token to detect shutdown. When Service Fabric requests cancellation, finish any work and exit `RunAsync` as quickly as possible.

Diagnostics. Application logs

Storage

Writing data to Azure Storage fails

Detection. The client receives errors when writing.

Recovery:

1. Retry the operation, to recover from transient failures. The [retry policy](#) in the client SDK handles this automatically.
2. Implement the Circuit Breaker pattern to avoid overwhelming storage.
3. If N retry attempts fail, perform a graceful fallback. For example:
 - Store the data in a local cache, and forward the writes to storage later, when the service becomes available.

- If the write action was in a transactional scope, compensate the transaction.

Diagnostics. Use [storage metrics](#).

Reading data from Azure Storage fails.

Detection. The client receives errors when reading.

Recovery:

1. Retry the operation, to recover from transient failures. The [retry policy](#) in the client SDK handles this automatically.
2. For RA-GRS storage, if reading from the primary endpoint fails, try reading from the secondary endpoint. The client SDK can handle this automatically. See [Azure Storage replication](#).
3. If N retry attempts fail, take a fallback action to degrade gracefully. For example, if a product image can't be retrieved from storage, show a generic placeholder image.

Diagnostics. Use [storage metrics](#).

Virtual machine

Connection to a backend VM fails.

Detection. Network connection errors.

Recovery:

- Deploy at least two backend VMs in an availability set, behind a load balancer.
- If the connection error is transient, sometimes TCP will successfully retry sending the message.
- Implement a retry policy in the application.
- For persistent or nontransient errors, implement the [Circuit Breaker](#) pattern.
- If the calling VM exceeds its network egress limit, the outbound queue will fill up. If the outbound queue is consistently full, consider scaling out.

Diagnostics. Log events at service boundaries.

VM instance becomes unavailable or unhealthy.

Detection. Configure a Load Balancer [health probe](#) that signals whether the VM instance is healthy. The probe should check whether critical functions are responding correctly.

Recovery. For each application tier, put multiple VM instances into the same availability set, and place a load balancer in front of the VMs. If the health probe fails, the Load Balancer stops sending new connections to the unhealthy instance.

Diagnostics. - Use Load Balancer [log analytics](#).

- Configure your monitoring system to monitor all of the health monitoring endpoints.

Operator accidentally shuts down a VM.

Detection. N/A

Recovery. Set a resource lock with `ReadOnly` level. See [Lock resources with Azure Resource Manager](#).

Diagnostics. Use [Azure Activity Logs](#).

WebJobs

Continuous job stops running when the SCM host is idle.

Detection. Pass a cancellation token to the WebJob function. For more information, see [Graceful shutdown](#).

Recovery. Enable the `Always On` setting in the web app. For more information, see [Run Background tasks with WebJobs](#).

Application design

Application can't handle a spike in incoming requests.

Detection. Depends on the application. Typical symptoms:

- The website starts returning HTTP 5xx error codes.
- Dependent services, such as database or storage, start to throttle requests. Look for HTTP errors such as HTTP 429 (Too Many Requests), depending on the service.
- HTTP queue length grows.

Recovery:

- Scale out to handle increased load.
- Mitigate failures to avoid having cascading failures disrupt the entire application.
Mitigation strategies include:
 - Implement the [Throttling pattern](#) to avoid overwhelming backend systems.
 - Use [queue-based load leveling](#) to buffer requests and process them at an appropriate pace.
 - Prioritize certain clients. For example, if the application has free and paid tiers, throttle customers on the free tier, but not paid customers. See [Priority queue pattern](#).

Diagnostics. Use [App Service diagnostic logging](#). Use a service such as [Azure Log Analytics](#), [Application Insights](#), or [New Relic](#) to help understand the diagnostic logs.



A sample is available [here](#). It uses [Polly](#) for these exceptions:

- 429 - Throttling
- 408 - Timeout
- 401 - Unauthorized
- 503 or 5xx - Service unavailable

One of the operations in a workflow or distributed transaction fails.

Detection. After N retry attempts, it still fails.

Recovery:

- As a mitigation plan, implement the [Scheduler Agent Supervisor](#) pattern to manage the entire workflow.
- Don't retry on timeouts. There is a low success rate for this error.
- Queue work, in order to retry later.

Diagnostics. Log all operations (successful and failed), including compensating actions. Use correlation IDs, so that you can track all operations within the same transaction.

A call to a remote service fails.

Detection. HTTP error code.

Recovery:

1. Retry on transient failures.
2. If the call fails after N attempts, take a fallback action. (Application specific.)
3. Implement the [Circuit Breaker pattern](#) to avoid cascading failures.

Diagnostics. Log all remote call failures.

Next steps

See [Resiliency and dependencies](#) in the Azure Well-Architected Framework. Building failure recovery into the system should be part of the architecture and design phases from the beginning to avoid the risk of failure.

Recover from a region-wide service disruption

Article • 03/20/2023

Azure is divided physically and logically into units called regions. A region consists of one or more data centers in close proximity. Many regions and services also support [availability zones](#), which can be used to provide more resiliency against outages in a single data center. Consider using regions with availability zones to improve the availability of your solution.

Under rare circumstances, it is possible that facilities in an entire availability zone or region can become inaccessible, for example, due to network failures. Or, facilities can be lost entirely, for example, due to a natural disaster. Azure has capabilities for creating applications that are distributed across zones and regions. Such distribution helps to minimize the possibility that a failure in one zone or region could affect other zones or regions.

Cloud services

Resource management

You can distribute compute instances across regions by creating a separate cloud service in each target region, and then publishing the deployment package to each cloud service. However, distributing traffic across cloud services in different regions must be implemented by the application developer or with a traffic management service.

Determining the number of spare role instances to deploy in advance for disaster recovery is an important aspect of capacity planning. Having a full-scale secondary deployment ensures that capacity is already available when needed; however, this effectively doubles the cost. A common pattern is to have a small, secondary deployment, just large enough to run critical services. This small secondary deployment is a good idea, both to reserve capacity, and for testing the configuration of the secondary environment.

Note

The subscription quota is not a capacity guarantee. The quota is simply a credit limit. To guarantee capacity, the required number of roles must be defined in the

service model, and the roles must be deployed.

Load Balancing

To load balance traffic across regions requires a traffic management solution. Azure provides [Azure Traffic Manager](#). You can also take advantage of third-party services that provide similar traffic management capabilities.

Strategies

Many alternative strategies are available for implementing distributed compute across regions. These must be tailored to the specific business requirements and circumstances of the application. At a high level, the approaches can be divided into the following categories:

- **Redeploy on disaster:** In this approach, the application is redeployed from scratch at the time of disaster. This is appropriate for non-critical applications that don't require a guaranteed recovery time.
- **Warm Spare (Active/Passive):** A secondary hosted service is created in an alternate region, and roles are deployed to guarantee minimal capacity; however, the roles don't receive production traffic. This approach is useful for applications that have not been designed to distribute traffic across regions.
- **Hot Spare (Active/Active):** The application is designed to receive production load in multiple regions. The cloud services in each region might be configured for higher capacity than required for disaster recovery purposes. Alternatively, the cloud services might scale out as necessary at the time of a disaster and fail over. This approach requires substantial investment in application design, but it has significant benefits. These include low and guaranteed recovery time, continuous testing of all recovery locations, and efficient usage of capacity.

A complete discussion of distributed design is outside the scope of this document. For more information, see [Disaster Recovery and High Availability for Azure Applications](#).

Virtual machines

Recovery of infrastructure as a service (IaaS) virtual machines (VMs) is similar to platform as a service (PaaS) compute recovery in many respects. There are important differences, however, due to the fact that an IaaS VM consists of both the VM and the VM disk.

- **Use Azure Backup to create cross region backups that are application consistent.** [Azure Backup](#) enables customers to create application consistent backups across multiple VM disks, and support replication of backups across regions. You can do this by choosing to geo-replicate the backup vault at the time of creation. Replication of the backup vault must be configured at the time of creation. It can't be set later. If a region is lost, Microsoft will make the backups available to customers. Customers will be able to restore to any of their configured restore points.
- **Separate the data disk from the operating system disk.** An important consideration for IaaS VMs is that you cannot change the operating system disk without re-creating the VM. This is not a problem if your recovery strategy is to redeploy after disaster. However, it might be a problem if you are using the Warm Spare approach to reserve capacity. To implement this properly, you must have the correct operating system disk deployed to both the primary and secondary locations, and the application data must be stored on a separate drive. If possible, use a standard operating system configuration that can be provided on both locations. After a failover, you must then attach the data drive to your existing IaaS VMs in the secondary DC. Use AzCopy to copy snapshots of the data disk(s) to a remote site.
- **Be aware of potential consistency issues after a geo-failover of multiple VM Disks.** VM Disks are implemented as Azure Storage blobs, and have the same geo-replication characteristic. Unless [Azure Backup](#) is used, there are no guarantees of consistency across disks, because geo-replication is asynchronous and replicates independently. Individual VM disks are guaranteed to be in a crash consistent state after a geo-failover, but not consistent across disks. This could cause problems in some cases (for example, in the case of disk striping).
- **Use Azure Site Recovery to replicate applications across regions.** With [Azure Site Recovery](#), customers don't need to worry about separating data disks from operating system disks or about potential consistency issues. Azure Site Recovery replicates workloads running on physical and virtual machines from a primary site (either on-premises or in Azure) to a secondary location (in Azure). When an outage occurs at the customer's primary site, a failover can be triggered to quickly return the customer to an operational state. After the primary location is restored, customers can then fail back. They can easily replicate using recovery points with application-consistent snapshots. These snapshots capture disk data, all in-memory data, and all in-process transactions. Azure Site Recovery allows customers to keep recovery time objectives (RTO) and recovery point objectives (RPO) within organizational limits. Customers can also easily run disaster recovery drills without affecting applications in production. Using recovery plans, customers

can sequence the failover and recovery of multitier applications running on multiple VMs. They can group machines together in a recovery plan, and optionally add scripts and manual actions. Recovery plans can be integrated with Azure Automation runbooks.

Storage

Recovery by using geo-redundant storage of blob, table, queue, and VM disk storage

In Azure, blobs, tables, queues, and VM disks are all geo-replicated by default. This is referred to as geo-redundant storage (GRS). GRS replicates storage data to a paired datacenter located hundreds of miles apart within a specific geographic region. GRS is designed to provide additional durability in case there is a major datacenter disaster. Microsoft controls when failover occurs, and failover is limited to major disasters in which the original primary location is deemed unrecoverable in a reasonable amount of time. Under some scenarios, this can be several days. Data is typically replicated within a few minutes, although synchronization interval is not yet covered by a service level agreement.

If a geo-failover occurs, there will be no change to how the account is accessed (the URL and account key will not change). The storage account will, however, be in a different region after failover. This could impact applications that require regional affinity with their storage account. Even for services and applications that do not require a storage account in the same datacenter, the cross-datacenter latency and bandwidth charges might be compelling reasons to move traffic to the failover region temporarily. This could factor into an overall disaster recovery strategy.

In addition to automatic failover provided by GRS, Azure has introduced a service that gives you read access to the copy of your data in the secondary storage location. This is called read-access geo-redundant storage (RA-GRS).

For more information about both GRS and RA-GRS storage, see [Azure Storage replication](#).

Geo-replication region mappings

It is important to know where your data is geo-replicated, in order to know where to deploy the other instances of your data that require regional affinity with your storage. For more information, see [Azure Paired Regions](#).

Geo-replication pricing

Geo-replication is included in current pricing for Azure Storage. This is called geo-redundant storage (GRS). If you do not want your data geo-replicated, you can disable geo-replication for your account. This is called locally redundant storage (LRS), and it is charged at a discounted price compared to GRS.

Determining if a geo-failover has occurred

If a geo-failover occurs, this will be posted to the [Azure Service Health Dashboard](#). Applications can implement an automated means of detecting this, however, by monitoring the geo-region for their storage account. This can be used to trigger other recovery operations, such as activation of compute resources in the geo-region where their storage moved to. You can perform a query for this from the service management API, by using [Get Storage Account Properties](#). The relevant properties are:

Console

```
<GeoPrimaryRegion>primary-region</GeoPrimaryRegion>
<StatusOfPrimary>[Available|Unavailable]</StatusOfPrimary>
<LastGeoFailoverTime>DateTime</LastGeoFailoverTime>
<GeoSecondaryRegion>secondary-region</GeoSecondaryRegion>
<StatusOfSecondary>[Available|Unavailable]</StatusOfSecondary>
```

Database

SQL Database

Azure SQL Database provides two types of recovery: geo-restore and active geo-replication.

Geo-restore

[Geo-restore](#) is also available with Basic, Standard, and Premium databases. It provides the default recovery option when the database is unavailable because of an incident in the region where your database is hosted. Similar to point-in-time restore, geo-restore relies on database backups in geo-redundant Azure storage. It restores from the geo-replicated backup copy, and therefore is resilient to the storage outages in the primary region. For more information, see [Restore an Azure SQL Database or failover to a secondary](#).

Active geo-replication

[Active geo-replication](#) is available for all database tiers. It's designed for applications that have more aggressive recovery requirements than geo-restore can offer. Using active geo-replication, you can create up to four readable secondaries on servers in different regions. You can initiate failover to any of the secondaries. In addition, active geo-replication can be used to support the application upgrade or relocation scenarios, as well as load balancing for read-only workloads. For details, see [Configure active geo-replication for Azure SQL Database and initiate failover](#). Refer to [Designing globally available services using Azure SQL Database](#) and [Managing rolling upgrades of cloud applications by using SQL Database active geo-replication](#) for details on how to design and implement applications and applications upgrade without downtime.

SQL Server on Azure Virtual Machines

A variety of options are available for recovery and high availability for SQL Server 2012 (and later) running in Azure Virtual Machines. For more information, see [High availability and disaster recovery for SQL Server in Azure Virtual Machines](#).

Other Azure platform services

When attempting to run your cloud service in multiple Azure regions, you must consider the implications for each of your dependencies. In the following sections, the service-specific guidance assumes that you must use the same Azure service in an alternate Azure datacenter. This involves both configuration and data-replication tasks.

Note

In some cases, these steps can help to mitigate a service-specific outage rather than an entire datacenter event. From the application perspective, a service-specific outage might be just as limiting and would require temporarily migrating the service to an alternate Azure region.

Service Bus

Azure Service Bus uses a unique namespace that does not span Azure regions. So the first requirement is to set up the necessary service bus namespaces in the alternate region. However, there are also considerations for the durability of the queued messages. There are several strategies for replicating messages across Azure regions.

For the details on these replication strategies and other disaster recovery strategies, see [Best practices for insulating applications against Service Bus outages and disasters](#).

App Service

To migrate an Azure App Service application, such as Web Apps or Mobile Apps, to a secondary Azure region, you must have a backup of the website available for publishing. If the outage does not involve the entire Azure datacenter, it might be possible to use FTP to download a recent backup of the site content. Then create a new app in the alternate region, unless you have previously done this to reserve capacity. Publish the site to the new region, and make any necessary configuration changes. These changes could include database connection strings or other region-specific settings. If necessary, add the site's SSL certificate and change the DNS CNAME record so that the custom domain name points to the redeployed Azure Web App URL.

HDInsight

The data associated with HDInsight is stored by default in Azure Blob Storage. HDInsight requires that a Hadoop cluster processing MapReduce jobs must be colocated in the same region as the storage account that contains the data being analyzed. Provided you use the geo-replication feature available to Azure Storage, you can access your data in the secondary region where the data was replicated if for some reason the primary region is no longer available. You can create a new Hadoop cluster in the region where the data has been replicated and continue processing it.

SQL Reporting

At this time, recovering from the loss of an Azure region requires multiple SQL Reporting instances in different Azure regions. These SQL Reporting instances should access the same data, and that data should have its own recovery plan in the event of a disaster. You can also maintain external backup copies of the RDL file for each report.

Media Services

Azure Media Services has a different recovery approach for encoding and streaming. Typically, streaming is more critical during a regional outage. To prepare for this, you should have a Media Services account in two different Azure regions. The encoded content should be located in both regions. During a failure, you can redirect the streaming traffic to the alternate region. Encoding can be performed in any Azure

region. If encoding is time-sensitive, for example during live event processing, you must be prepared to submit jobs to an alternate datacenter during failures.

Virtual network

Configuration files provide the quickest way to set up a virtual network in an alternate Azure region. After configuring the virtual network in the primary Azure region, [export the virtual network settings](#) for the current network to a network configuration file. If an outage occurs in the primary region, [restore the virtual network](#) from the stored configuration file. Then configure other cloud services, virtual machines, or cross-premises settings to work with the new virtual network.

There are VNET related resources which we need to take into account (Ex. NSG, DNS, Route Tables). The method described in [Infrastructure as a code](#) is a way to generate the same environment every time, and you can deploy in a new region.

Checklists for disaster recovery

Cloud Services checklist

1. Review the Cloud Services section of this document.
2. Create a cross-region disaster recovery strategy.
3. Understand trade-offs in reserving capacity in alternate regions.
4. Use traffic routing tools, such as Azure Traffic Manager.

Virtual Machines checklist

1. Review the Virtual Machines section of this document.
2. Use [Azure Backup ↗](#) to create application consistent backups across regions.

Storage checklist

1. Review the Storage section of this document.
2. Do not disable geo-replication of storage resources.
3. Understand alternate region for geo-replication if a failover occurs.
4. Create custom backup strategies for user-controlled failover strategies.

SQL Database checklist

1. Review the SQL Database section of this document.

2. Use [Geo-restore](#) or [geo-replication](#) as appropriate.

SQL Server on Virtual Machines checklist

1. Review the SQL Server on Virtual Machines section of this document.
2. Use cross-region AlwaysOn Availability Groups or database mirroring.
3. Alternately use backup and restore to blob storage.

Service Bus checklist

1. Review the Service Bus section of this document.
2. Configure a Service Bus namespace in an alternate region.
3. Consider custom replication strategies for messages across regions.

App Service checklist

1. Review the App Service section of this document.
2. Maintain site backups outside of the primary region.
3. If outage is partial, attempt to retrieve current site with FTP.
4. Plan to deploy the site to new or existing site in an alternate region.
5. Plan configuration changes for both application and DNS CNAME records.

HDInsight checklist

1. Review the HDInsight section of this document.
2. Create a new Hadoop cluster in the region with replicated data.

SQL Reporting checklist

1. Review the SQL Reporting section of this document.
2. Maintain an alternate SQL Reporting instance in a different region.
3. Maintain a separate plan to replicate the target to that region.

Media Services checklist

1. Review the Media Services section of this document.
2. Create a Media Services account in an alternate region.
3. Encode the same content in both regions to support streaming failover.
4. Submit encoding jobs to an alternate region if a service disruption occurs.

Virtual Network checklist

1. Review the Virtual Network section of this document.
2. Use exported virtual network settings to re-create it in another region.

Recommendations for designing for redundancy

Article • 11/15/2023

Applies to this Azure Well-Architected Framework Reliability checklist recommendation:

[+] Expand table

RE:05 Add redundancy at different levels, especially for critical flows. Apply redundancy to the compute, data, network, and other infrastructure tiers in accordance with the identified reliability targets.

Related guides: [Highly available multiregional design](#) | [Using availability zones and regions](#)

This guide describes the recommendations for adding redundancy throughout critical flows at different workload layers, which optimizes resiliency. Meet the requirements of your defined reliability targets by applying the proper levels of redundancy to your compute, data, networking, and other infrastructure tiers. Apply this redundancy to give your workload a strong, reliable foundation to build on. When you build your workload without infrastructure redundancy, there's a high risk of extended downtime due to potential failures.

Definitions

[+] Expand table

Term	Definition
Redundancy	The implementation of multiple identical instances of a workload component.
Polyglot persistence	The concept of using different storage technologies by the same application or solution to take advantage of the best capabilities of each component.
Data consistency	The measure of how in sync or out of sync a given dataset is across multiple stores.
Partitioning	The process of physically dividing data into separate data stores.
Shard	A horizontal database partitioning strategy that supports multiple storage instances with a common schema. Data isn't replicated in all instances.

Key design strategies

In the context of reliability, use redundancy to contain problems that affect a single resource and ensure that those problems don't affect the reliability of the entire system. Use the information that you identified about your critical flows and reliability targets to make informed decisions that are required for each flow's redundancy.

For example, you might have multiple web server nodes running at once. The criticality of the flow that they support might require that all of them have replicas that are ready to accept traffic if there's a problem that affects the entire pool, for example a regional outage. Alternatively, because large-scale problems are rare and it's costly to deploy an entire set of replicas, you might deploy a limited number of replicas so the flow operates in a degraded state until you resolve the problem.

When you design for redundancy in the context of performance efficiency, distribute the load across multiple redundant nodes to ensure that each node performs optimally. In the context of reliability, build in spare capacity to absorb failures or malfunctions that affect one or more nodes. Ensure that the spare capacity can absorb failures for the entire time that's needed to recover the affected nodes. With this distinction in mind, both strategies need to work together. If you spread traffic across two nodes for performance and they both run at 60 percent utilization and one node fails, your remaining node is at risk of becoming overwhelmed because it can't operate at 120 percent. Spread the load out with another node to ensure that your performance and reliability targets are upheld.



Tradeoffs:

- More workload redundancy equates to more costs. Carefully consider adding redundancy and regularly review your architecture to ensure that you're managing costs, especially when you use overprovisioning. When you use overprovisioning as a resiliency strategy, balance it with a well-defined [scaling strategy](#) to minimize cost inefficiencies.
- There can be performance tradeoffs when you build in a high degree of redundancy. For example, resources that spread across availability zones or regions can affect performance because you have to send traffic over high-latency connections between redundant resources, like web servers or database instances.
- Different flows within the same workload might have different reliability requirements. Flow-specific redundancy designs can potentially introduce

complexity into the overall design.

Redundant architecture design

Consider two approaches when you design a redundant architecture: active-active or active-passive. Choose your approach depending on the criticality of the user flow and system flow that the infrastructure components support. In terms of reliability, a multi-region active-active design helps you achieve the highest level of reliability possible, but it's significantly more expensive than an active-passive design. Deciding the appropriate geographic regions become the next critical choice. You can also use these design approaches for a single region by using availability zones. For more information, see [Recommendations for highly available multi-region design](#).

Deployment stamps and units of scale

Whether you deploy in an active-active or active-passive model, follow the [Deployment Stamps design pattern](#) to ensure that you deploy your workload in a repeatable, scalable way. Deployment stamps are the groupings of resources that are required to deliver your workload to a given subset of your customers. For example, the subset might be a regional subset or a subset with all the same data privacy requirements as your workload. Think of each stamp as a *unit of scale* that you can duplicate to scale your workload horizontally or to perform blue-green deployments. Design your workload with deployment stamps to optimize your active-active or active-passive implementation for resiliency and management burden. Planning for multi-region scale out is also important to overcome potential temporary resource capacity constraints in a region.

Availability zones within Azure regions

Whether you deploy an active-active or an active-passive design, take advantage of [availability zones](#) within the active regions to fully optimize your resiliency. Many Azure regions provide multiple availability zones, which are separated groups of data centers within a region. Depending on the Azure service, you can take advantage of availability zones by deploying elements of your workload redundantly across zones or pinning elements to specific zones. For more information, see [Recommendations for using availability zones and regions](#).

Implement zone redundancy for compute resources

- Choose the appropriate [compute service](#) for your workload. Depending on the type of workload that you design, there might be several options available. Research the available services and understand which types of workloads work best on a given compute service. For example, SAP workloads are typically best suited for infrastructure as a service (IaaS) compute services. For a containerized application, determine the specific functionality you need to have control over to determine whether to use Azure Kubernetes Service (AKS) or a platform as a service (PaaS) solution. Your cloud platform fully manages a PaaS service.
- Use PaaS compute options if your requirements allow it. Azure fully manages PaaS services, which reduces your management burden, and a documented degree of redundancy is built in.
- Use Azure Virtual Machine Scale Sets if you need to deploy virtual machines (VMs). With Virtual Machine Scale Sets, you can automatically spread your compute evenly across availability zones.
- Keep your compute layer *clean of any state* because individual nodes that serve requests might be deleted, faulted, or replaced at any time.
- Use zone-redundant services where possible to provide higher resilience without increasing your operational burden.
- Overprovision critical resources to mitigate failures of redundant instances, even before autoscaling operations begin, so the system continues to operate after a component failure. Calculate the acceptable effect of a fault when you incorporate overprovisioning into your redundancy design. As with your redundancy decision-making process, your reliability targets and financial tradeoff decisions determine the extent that you add spare capacity with overprovisioning. Overprovisioning specifically refers to *scaling out*, which means adding extra instances of a given compute resource type, rather than increasing the compute capabilities of any single instance. For example, if you change a VM from a lower-tier SKU to a higher-tier SKU.
- Deploy IaaS services manually or via automation in each availability zone or region in which you intend to implement your solution. Some PaaS services have built-in capabilities that are automatically replicated across availability zones and regions.

Implement zone redundancy for data resources

- Determine whether synchronous or asynchronous data replication is necessary for your workload's functionality. To help you make this determination, see [Recommendations for using availability zones and regions](#).

- Consider the growth rate of your data. For capacity planning, plan for data growth, data retention, and archiving to ensure your reliability requirements are met as the amount of data in your solution increases.
- Distribute data geographically, as supported by your service, to minimize the effect of geographically localized failures.
- Replicate data across geographical regions to provide resilience to regional outages and catastrophic failures.
- Automate failover after a database instance failure. You can configure automated failover in multiple Azure PaaS data services. Automated failover isn't required for data stores that support multi-region writes, like Azure Cosmos DB. For more information, see [Recommendations for designing a disaster recovery strategy](#).
- Use the best [data store](#) for your data. Embrace polyglot persistence or solutions that use a mix of data store technologies. Data includes more than just persisted application data. It also includes application logs, events, messages, and caches.
- Consider data consistency requirements and use [eventual consistency](#) when requirements allow it. When data is distributed, use appropriate coordination to enforce strong consistency guarantees. Coordination can reduce your throughput and make your systems tightly coupled, which can make them more brittle. For example, if an operation updates two databases, instead of putting it into a single transaction scope, it's better if the system can accommodate eventual consistency.
- Partition data for availability. [Database partitioning](#) improves scalability and it can also improve availability. If one shard goes down, the other shards are still reachable. A failure in one shard only disrupts a subset of the total transactions.
- If sharding isn't an option, you can use the [Command and Query Responsibility Segregation \(CQRS\) pattern](#) to separate your read-write and read-only data models. Add more redundant read-only database instances to provide more resilience.
- Understand the built-in replication and redundancy capabilities of the stateful platform services that you use. For specific redundancy capabilities of stateful data services, see [Related links](#).

Implement zone redundancy for networking resources

- Decide on a reliable and scalable network topology. Use a hub-and-spoke model or an Azure Virtual WAN model to help you organize your cloud infrastructure in

logical patterns that make your redundancy design easier to build and scale.

- Select the appropriate [network service](#) to balance and redirect requests within or across regions. Use global or zone-redundant load balancing services when possible to meet your reliability targets.
- Ensure that you have allocated sufficient IP address space in your virtual networks and subnets to account for planned usage, including scale-out requirements.
- Ensure that the application can scale within the port limits of the chosen application hosting platform. If an application initiates several outbound TCP or UDP connections, it might exhaust all available ports and cause poor application performance.
- Choose SKUs and configure networking services that can meet your bandwidth and availability requirements. A VPN gateway's throughput varies based on their SKU. Support for zone redundancy is only available for some load balancer SKUs.
- Ensure that your design for handling DNS is built with a focus on resilience and supports redundant infrastructure.

Azure facilitation

The Azure platform helps you optimize the resiliency of your workload and add redundancy by:

- Providing built-in redundancy with many PaaS and software as a service (SaaS) solutions, some of which are configurable.
- Allowing you to design and implement intra-region redundancy by using [availability zones](#) and inter-region redundancy.
- Offering replica-aware load balancing services like [Azure Application Gateway](#), [Azure Front Door](#), and [Azure Load Balancer](#).
- Offering easily implemented geo-replication solutions like [active geo replication](#) for Azure SQL Database. Implement [global distribution](#) and transparent replication by using Azure Cosmos DB. Azure Cosmos DB offers two options for [handling conflicting writes](#). Choose the best option for your workload.
- Offering point-in-time restore capabilities for many PaaS data services.
- Mitigating port exhaustion via [Azure NAT Gateway](#) or [Azure Firewall](#).

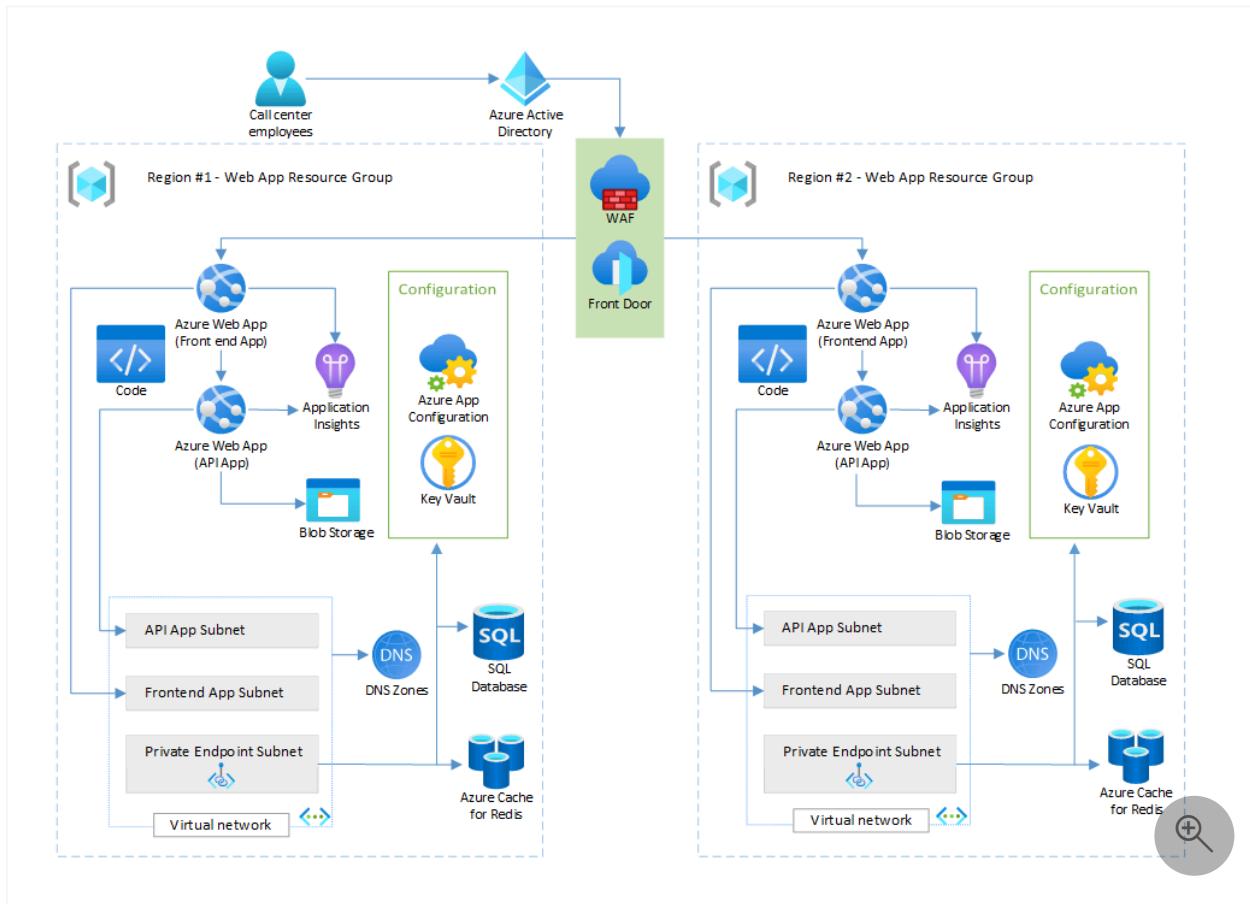
DNS-specific Azure facilitation

- For internal name resolution scenarios, use Azure DNS private zones, which have built-in zone redundancy and geo redundancy. For more information, see [Azure DNS private zone resiliency](#).
- For external name resolution scenarios, use Azure DNS public zones, which have built-in zone redundancy and geo redundancy.
- The public and private Azure DNS services are global services that are resilient to regional outages because zone data is globally available.
- For hybrid name resolution scenarios between on-premises and cloud environments, use Azure DNS Private Resolver. This service supports zone redundancy if your workload is located in a region that supports availability zones. A zone-wide outage requires no action during zone recovery. The service automatically self-heals and rebalances to take advantage of the healthy zone. For more information, see [Resiliency in Azure DNS Private Resolver](#).
- To eliminate a single point of failure and achieve a more resilient hybrid name resolution across regions, deploy two or more Azure DNS private resolvers across different regions. DNS failover, in a conditional forwarding scenario, is achieved by assigning a resolver as your primary DNS server. Assign the other resolver in a different region as a secondary DNS server. For more information, see [Set up DNS failover by using private resolvers](#).

Example

For an example of a multi-region redundant deployment, see [Baseline highly available zone-redundant web application](#).

The following diagram shows another example:



Related links

To learn more about redundancy, see the following resources:

- [Azure regions guide](#)
- [Azure Storage redundancy](#)
- [Zone-redundant storage](#)
- [Azure SQL Database active geo-replication](#)
- [Configure replication between two managed instances](#)

Reliability checklist

Refer to the complete set of recommendations.

[Reliability checklist](#)

Feedback

Was this page helpful?

[Yes](#)

[No](#)

Use infrastructure as code to update Azure landing zones

Article • 03/31/2023

This article describes the benefits of using infrastructure as code (IaC) to update Azure landing zones. Organizations need to update their landing zones as they operate to ensure that configurations are correct and they respond to the need for changes.

IaC can manage the whole life cycle, and it excels at managing the resources that it deploys. Organizations should plan to deploy their Azure landing zones with IaC. It requires planning to align existing non-IaC resources with IaC resources that are backed with state management. You need to map the existing resources to the desired state.

For more information, see [Keep your Azure landing zone up to date](#).

How infrastructure as code works

IaC refers to the practice and tools for managing the lifecycle of infrastructure resources by using machine-readable definition files. The definition for the infrastructure is written, versioned, deployed through pipelines, and then it becomes a part of the deployment for workloads.

IaC technologies are *declarative*, which means when IaC runs, it sets the configuration to what's described in the code, regardless of its current state. When you configure infrastructure through scripts, such as the Azure CLI or Azure PowerShell, they're *imperative*. Imperative scripts perform a set of actions, and the result depends on the current state plus the state after the actions.

So, if you have an infrastructure as code definition for an Azure resource, you can run that definition as often as you want, and it only creates a change if:

- The definition changes to add new resources, remove resources previously deployed, or modifies resources that were previously deployed.
- The deployed resource drifts from the configuration to reset the configuration to the defined one.

You can use IaC to restore the state by removing resources that are no longer needed and managing the lifecycle of resources through many changes.

Note

The specific mechanics to remove resources with IaC varies. For example, Azure Bicep requires the use of a `complete` deployment type to remediate out of scope resources. This command only works in specific scopes. For Terraform, resources have a `lifecycle` meta-argument that provides instructions for how Terraform should handle resources.

For Azure landing zones, there are two main options for infrastructure as code:

- Azure Bicep, which is a domain-specific language that's used to deploy Microsoft developed Azure resources. For more information, see [Azure landing zones - Bicep modules design considerations](#).
- Terraform, a product produced by Hashicorp, to deploy infrastructure to the cloud and on-premises. Terraform has specific Microsoft produced resource providers for the deployment of Azure resources. For more information, see [Azure landing zones - Terraform module design considerations](#).

The benefits of updating ALZ with infrastructure as code

The following benefits describe why you should use infrastructure as code to make your landing zone updates.

Reduce effort

It takes less effort to use infrastructure as code to perform updates compared to making manual changes. The IaC deployment helps answer the following questions:

- How are resources configured today?
- How will it be configured by this update?
- What changes will be made to bring it in line with this update?

When an infrastructure as code toolset runs, it can produce a comparison or "differential" readout of the changes. Review this readout before you commit changes to the environment.

The toolset can compile the information for the change rather than an operator or an engineer.

Reduce error

Due to the programmatic nature of the deployments, infrastructure as code reduces human error while it makes changes. It only changes what's defined, and it has preview options, so it reduces outages that are caused by failed or incomplete changes. It also has improved testing options.

Version control and history

Infrastructure as code deployments are backed by a definition file, so you can use source control to manage the versions of your definitions. Depending on the method of IaC that you use, you can reference the deployments in Azure for Bicep or your state file for Terraform to review the history of previous deployments.

When you use source control practices, it creates a new branch of your IaC to add changes and revisions. The branch's history in your source control system captures the iterations and changes. You can use it to deploy changes to a test environment until you're ready to merge and deploy the changes to production. For more information, see [Testing approach for Azure landing zones](#). Throughout this cycle, the deployment records capture the version that's used and the resources that are deployed, which provides a highly visible history.

Use these testing methods with Bicep for general testing purposes. With these methods, you can perform testing before you deploy the code, and you can test in non-production environments from your branch.

Testing environments

IaC deployments are repeatable, so you can use the same definition to deploy a second (or more) environment based on the deployment. This method is valuable for testing changes.

For example, if you want to replace your Azure Firewall by using the Premium SKU, you can deploy a test environment and validate the changes without changing production.

Catch configuration drifts

IaC provides a unique option to catch configuration drifts during updates. The deployment catches changes to the definition file and presents instances where the resource configuration differs from the definition.

Landing zone updates with IaC can help you catch this configuration drift and allow you to update the code appropriately, address these misconfigurations via the update, or address them in another way.

When you make a change to resources via the portal, CLI, or a non-IaC method, the change is implemented. The next time you run a deployment through IaC, it flags the comparison between the code-defined state and the actual state in the portal by using what-if or plan functions. Use this method to identify if an environment is modified outside of the code file.

After the misalignment is identified, you can run IaC to attempt to align the deployment with the definition. Use this method to identify issues and remediate scenarios depending on the nature of the issues, the nature of the run, and how the changes were made. For example, Terraform attempts to restore the baseline to resources it has deployed, and a `Complete` mode deployment in Bicep removes resources in a resource group that aren't part of the definition. These tools detect and repair configuration drift, but they might not address all issues.

For more information, see [Out-of-band changes](#) and [Detecting and managing drift with Terraform ↗](#).

Changes that are defined in the portal are cumbersome to implement back in to IaC. You must update the code to match the current state, which often involves reviewing each resource change and updating its parameters to match the "as is" configuration.

If you use IaC to manage your landing zone or other resources, you should only make changes outside of IaC as part of an emergency. Take precautions with accounts that have access to make changes directly, such as Privileged Identity Management.

Review general automation and security practices in the following articles:

- [Security Baseline discipline overview](#)
- [Identity Baseline discipline overview](#)
- [Operational compliance recommendations](#)
- [Platform automation design recommendations](#)

Next steps

Explore an introduction to the IaC tools in the following articles:

- [What is Bicep?](#)
- [What is Terraform? ↗](#)
- [Testing Terraform code](#)

IT management and operations in the cloud

Article • 12/01/2022

As a business moves to a cloud-based model, the importance of proper management and operations can't be overstated. Unfortunately, few organizations are prepared for the IT management shift that's required for success in building a cloud-first operating model. This section of the Cloud Adoption Framework outlines the operating model, processes, and tooling that have proven successful in the cloud. These areas represent a minor but fundamental change in the way the business should view IT operations and management as it begins to adopt the cloud.

Brief history of IT management

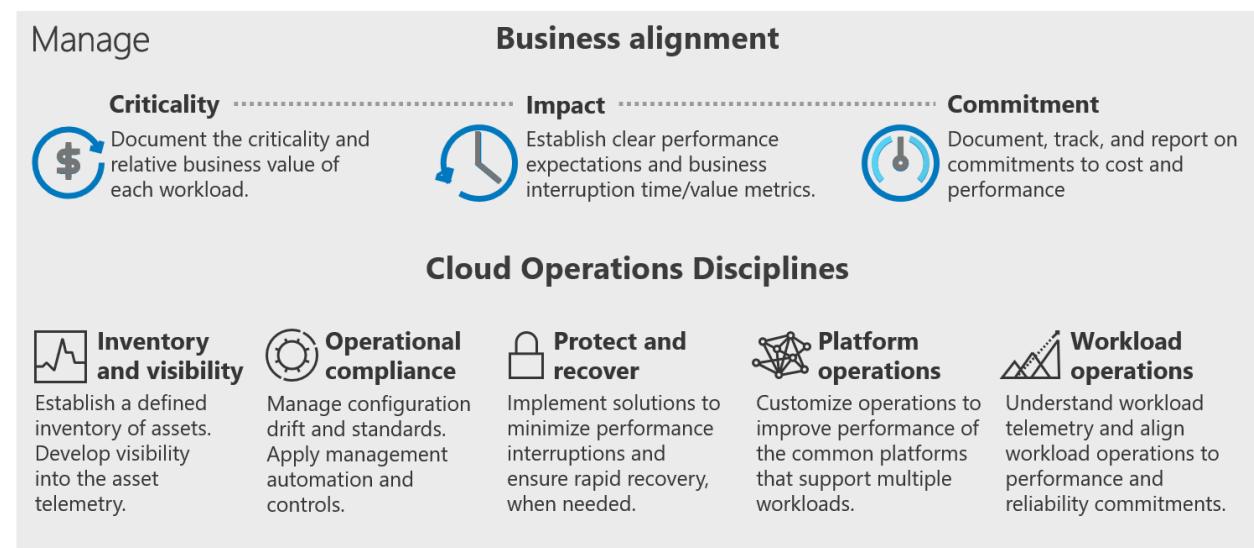
Before the cloud, IT management grew from a simple acquisition function. Acquisition of technical equipment to support business processes required technical expertise and deep experience with a specific group of equipment vendors. IT management consolidated the selection, acquisition, and configuration of IT assets. Generally, the acquired assets included storage, computing power, networking, and other similar assets that are required to power the desired business function. As the primary subject matter experts on the equipment, IT was also tasked with operating the equipment to ensure maximum performance and minimal business disruptions.

When the business builds out new technology solutions, it has a clear need that can justify the significant expenses associated with acquiring assets, or even building out full datacenters. When it builds solutions, the business sees the acquisition costs as an investment in the future. After the business need is met, the perception of the same costs shifts. Costs that are associated with existing solutions are seen as operational drag that's created by past needs. That perception is why many businesses view IT as a cost center. It's also why many IT organizations experience regular cost-control exercises or reductions in IT staff.

Cloud management

The historical IT operating model was sufficient for over 20 years. But that model is now outdated and is less desirable than cloud-first alternatives. When IT management teams move to the cloud, they have an opportunity to rethink this model and drive greater

value for the business. This article series outlines a modernized model of IT management.



Next steps

For a deeper understanding of the new cloud management model, start by [understanding business alignment](#).

[Understand business alignment](#)

Create business alignment in cloud management

Article • 12/01/2022

In on-premises environments, IT assets (applications, virtual machines, VM hosts, disk, servers, devices, and data sources) are managed by IT to support workload operations. In IT terms, a workload is a collection of IT assets that support a specific business operation. To help support business operations, IT management delivers processes that are designed to minimize disruptions to those assets. When an organization moves to the cloud, management and operations shift a bit, creating an opportunity to develop tighter business alignment.

Business vernacular

The first step in creating business alignment is to ensure term alignment. IT management, like most engineering professions, has amassed a collection of jargon, or highly technical terms. Such terms can lead to confusion for business stakeholders and make it difficult to map management services to business value.

Fortunately, the process of developing a cloud adoption strategy and cloud adoption plan creates an ideal opportunity to remap these terms. The process also creates opportunities to rethink commitments to operational management, in partnership with the business. The following article series walks you through this new approach across three specific terms that can help improve conversations among business stakeholders:

- **Criticality:** Mapping workloads to business processes. Ranking criticality to focus investments.
- **Impact:** Understanding the impact of potential outages to aid in evaluating return on investment for cloud management.
- **Commitment:** Developing true partnerships, by creating and documenting agreements with the business.

Note

Underlying these terms are classic IT terms such as SLA, RTO, and RPO. For more information on mapping specific business and IT terms, see **Business commitment in cloud management**.

Operations management workbook

To help capture decisions that result from this conversation about term alignment, an [operations management workbook ↗](#) is available on our GitHub site. This workbook does not perform SLA or cost calculations. It serves only to help capture such measures and forecast return on loss-avoidance efforts.

Alternatively, these same workloads and associated assets could be tagged directly in Azure, if the solutions are already deployed to the cloud.

Next steps

Start creating business alignment by defining [workload criticality](#).

[Define workload criticality](#)

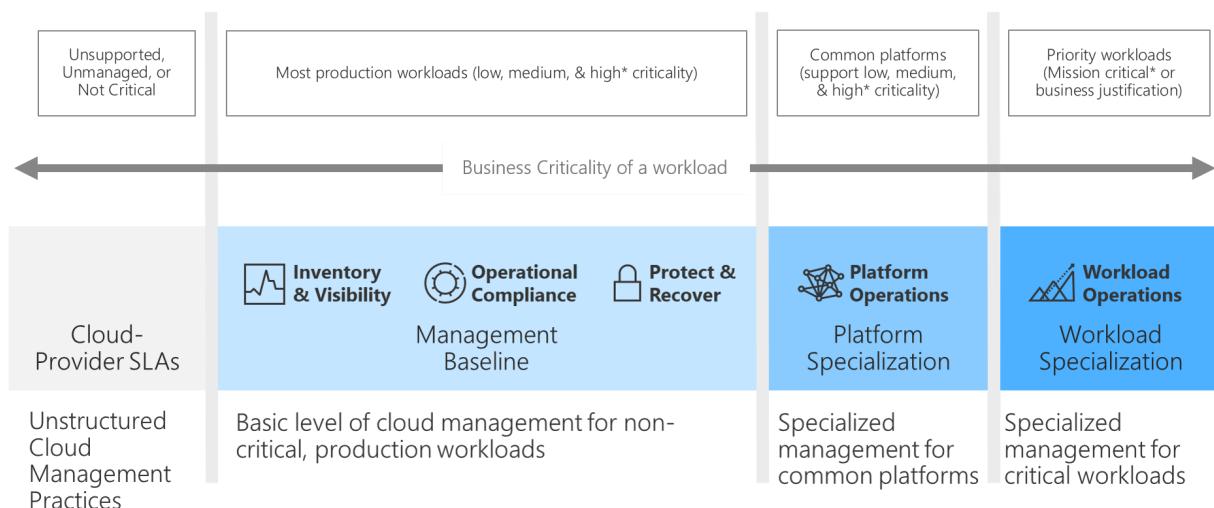
Business criticality in cloud management

Article • 04/28/2023

Across every business, there exist a small number of workloads that are too important to fail. These workloads are considered mission critical. When those workloads experience outages or performance degradation, the adverse impact on revenue and profitability can be felt across the entire company.

At the other end of the spectrum, some workloads can go months at a time without being used. Poor performance or outages for those workloads is not desirable, but the impact is isolated and limited.

Understanding the criticality of each workload in the IT portfolio is the first step toward establishing mutual commitments to cloud management. The following diagram illustrates a common alignment between the criticality scale to follow and the standard commitments made by the business.



Criticality scale

The first step in any business criticality alignment effort is to create a criticality scale. The following table presents a sample scale to be used as a reference, or template, for creating your own scale.

Criticality	Business view
Mission-critical	Affects the company's mission and might noticeably affect corporate profit-and-loss statements.

Criticality	Business view
Unit-critical	Affects the mission of a specific business unit and its profit-and-loss statements.
High	Might not hinder the mission, but affects high-importance processes. Measurable losses can be quantified in the case of outages.
Medium	Impact on processes is likely. Losses are low or immeasurable, but brand damage or upstream losses are likely.
Low	Impact on business processes isn't measurable. Neither brand damage nor upstream losses are likely. Localized impact on a single team is likely.
Unsupported	No business owner, team, or process that's associated with this workload can justify any investment in the ongoing management of the workload.

It's common for businesses to include additional criticality classifications that are specific to their industry, vertical, or specific business processes. Examples of additional classifications include:

- **Compliance-critical:** In heavily regulated industries, some workloads might be critical as part of an effort to maintain compliance requirements.
- **Security-critical:** Some workloads might not be mission critical, but outages could result in loss of data or unintended access to protected information.
- **Safety-critical:** When lives or the physical safety of employees and customers is at risk during an outage, it can be wise to classify workloads as safety-critical.
- **Sustainability-critical:** If your business focuses on the sustainability of some of its systems, consider this as a classification.

Importance of accurate criticality

Later in the cloud-adoption process, the cloud management team will use this classification to determine the amount of effort required to meet aligned levels of criticality. In on-premises environments, operations management is often purchased centrally and treated as a necessary business burden, with little or no additional operating costs. Like all cloud services, operations management is purchased on a per-asset basis as monthly operating costs.

Because there's a clear and direct cost to operations management in the cloud, it's important to properly align costs and desired criticality scales.

Select a default criticality

An initial review of every workload in the portfolio can be time consuming. To ensure that this effort doesn't block your broader cloud strategy, we recommend that your teams agree on a default criticality to apply to all workloads.

Based on the preceding criticality-scale table, we recommend that you adopt *medium* criticality as the default. Doing so will allow your cloud strategy team to quickly identify workloads that require a higher level of criticality.

Review operational compliance requirements

Understanding, identifying and designating business criticality is the first major step. The next management consideration topic is to ensure your operational compliance requirements, especially security compliances, are aligned. Performing the same methodologies for business criticality to your security compliance considerations will:

- Reveal the complexities of your business systems and their compliance requirements, such as sovereignty, industry, or privacy.
- Map any interdependencies with your development, operations, and security teams, ensuring that key point of contacts and subject matter experts are identified and well known across your organization.
- Ensure that your compliance adheres to any compliance audit and reporting requirements.

Apply the [operational compliance](#) content to ensure that your business requirements, criticality, and compliance requirements are aligned.

Next, review and use the [overview of the Azure Security Benchmark](#) content to ensure the various control domains are identified and mapped to your operational-compliance.

Lastly, ensure the incorporation of the governance and security content in [Security control v2: Governance and strategy](#) is documented within your overall business operational methodologies.

Use the template

The following steps apply if you're using the [operations management workbook](#) to plan for cloud management.

1. Record the criticality scale in the `Scale` worksheet.
2. Update each workload in either the `Example` worksheet or the `Clean Template` worksheet to reflect the default criticality in the `Criticality` column.

3. The business should enter the correct values to reflect any deviations from the default criticality.

Next steps

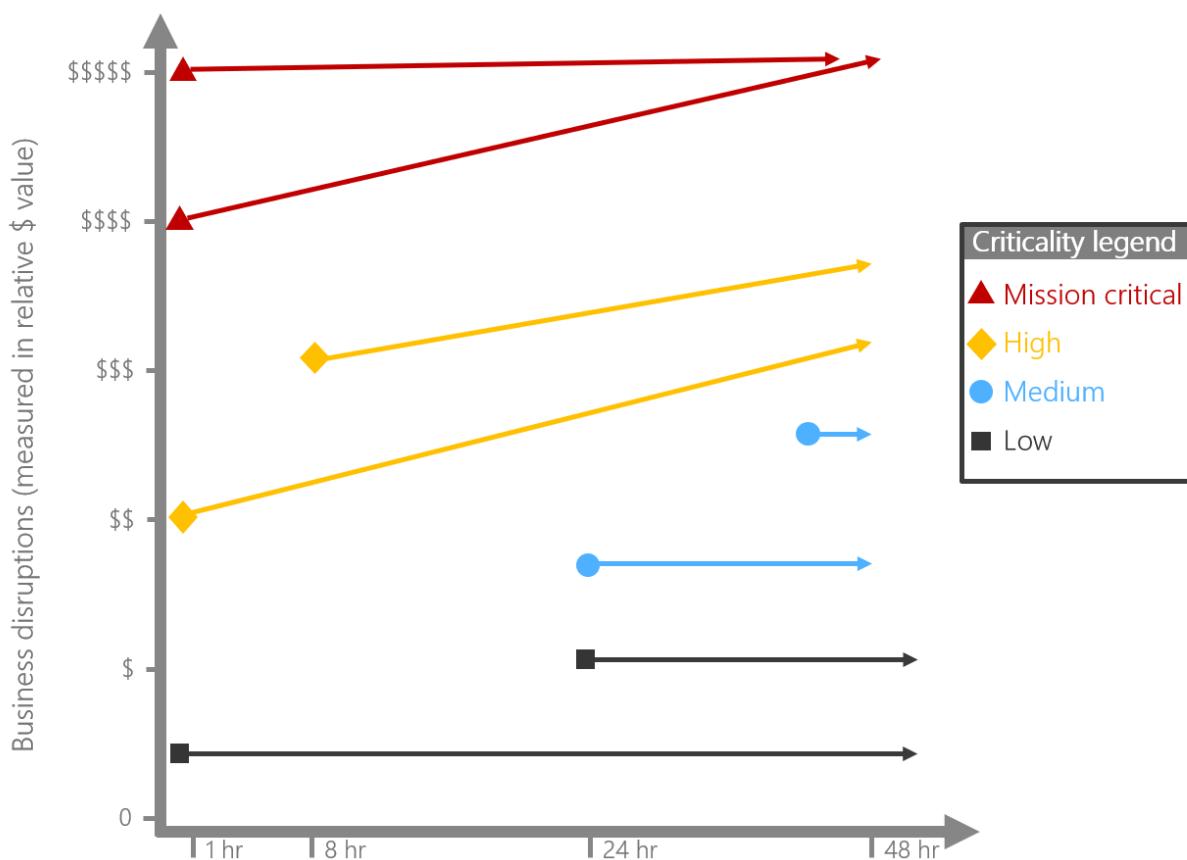
After your team has defined business criticality, you can [calculate and record business impact](#).

[Calculate and record business impact](#)

Business impact in cloud management

Article • 12/01/2022

Assume the best, prepare for the worst. In IT management, it's safe to assume that the workloads required to support business operations will be available and will perform within agreed-upon constraints, based on the selected criticality. However, to manage investments wisely, it's important to understand the impact on the business when an outage or performance degradation occurs. This importance is illustrated in the following graph, which maps potential business interruptions of specific workloads to the business impact of outages across a relative value scale.



To create a fair basis of comparison for the impact on various workloads across a portfolio, a time/value metric is suggested. The time/value metric captures the adverse impact of a workload outage. Generally, this impact is recorded as a direct loss of revenue or operating revenue during a typical outage period. More specifically, it calculates the amount of lost revenue for a unit of time. The most common time/value metric is *Impact per hour*, which measures operating revenue losses per hour of outage.

A few approaches can be used to calculate impact. You can apply any of the options in the following sections to achieve similar outcomes. It's important to use the same approach for each workload when you calculate protected losses across a portfolio.

Start with estimates

Current operating models might make it difficult to determine an accurate impact. Fortunately, few systems need a highly accurate loss calculation. In the previous step, *Classify Criticality*, we suggested that you start all workloads with a default of *medium criticality*. Medium criticality workloads generally receive a standard level of management support with a relatively low impact on operating cost. Only when a workload requires additional operational management resources might you require an accurate financial impact.

For all standardized workloads, business impact serves as a prioritization variable when you're recovering systems during an outage. Outside of those limited situations, the business impact creates little to no change in the operations management experience.

Calculate time

Depending on the nature of the workload, you could calculate losses differently. For high-paced transactional systems such as a real-time trading platform, losses per millisecond might be significant. Less frequently used systems, such as payroll, might not be used every hour. Whether the frequency of usage is high or low, it's important to normalize the time variable when you calculate financial impact.

Calculate total impact

When you want to consider additional management investments, it's more important that the business impact be more accurate. The following three approaches to calculating losses are ordered from most accurate to least accurate:

- **Adjusted losses:** If your business has experienced a major loss event in the past, such as a hurricane or other natural disaster, a claims adjuster might have calculated actual losses during the outage. These calculations are based on insurance industry standards for loss calculation and risk management. Using adjusted losses as the total amount of losses in a specific time frame can lead to highly accurate projections.
- **Historical losses:** If your on-premises environment has suffered historically from outages resulting from infrastructure instability, it can be a bit harder to calculate losses. But you can still apply the adjuster formulas used internally. To calculate historical losses, compare the deltas in sales, gross revenue, and operating costs across three time frames: before, during, and after outage. By examining these deltas, you can identify accurate losses when no other data is available.

- **Complete loss calculation:** If no historical data is available, you can derive a comparative loss value. In this model, you determine the average gross revenue per hour for the business unit. When you're projecting loss avoidance investments, it's not fair to assume that a complete system outage equates to a 100 percent loss of revenue. But you can use this assumption as a rough basis for comparing loss impacts and prioritizing investments.

Before you make certain assumptions about potential losses associated with workload outages, it's a good idea to work with your finance department to determine the best approach to such calculations.

Calculate workload impact

When you're calculating losses by applying historical data, you might have enough information to clearly determine the contribution of each workload to those losses. Performing this evaluation is where partnerships within the business are absolutely critical. After the total impact has been calculated, that impact must be attributed across each of the workloads. That distribution of impact should come from the business stakeholders, who should agree on the relative and cumulative impact of each workload. To that end, your team should solicit feedback from business executives to validate alignment. Such feedback is often equal parts emotion and subject matter expertise. It's important that this exercise represent the logic and beliefs of the business stakeholders who should have a say in budget allocation.

Use the template

If you're using the [operations management workbook](#) to plan for cloud management, consider doing the following:

- Each business should update each workload in either the `Example` worksheet or the `Clean Template` worksheet, along with the `Time/Value Impact` of each workload. By default, `Time/Value Impact` represents the projected losses per hour associated with an outage of the workload.

Next steps

After the business has defined impact, you can [align commitments](#).

[Align management commitments with the business](#)

Business commitment in cloud management

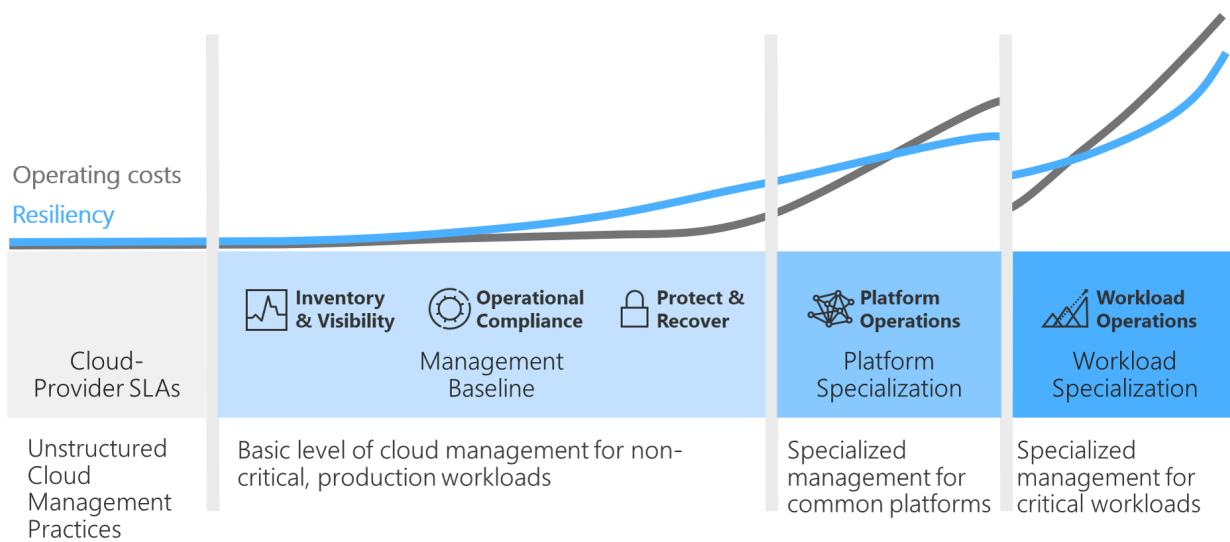
Article • 09/24/2024

A *business commitment* helps you define your level of operational management at an acceptable operating cost. To define a business commitment, you must balance priorities. This article describes how to evaluate data points and calculations to find that balance.

You can have commitments that are related to business stability that justify business decisions. Stability commitments can include service-level agreements (SLAs) or a certain level of technical resiliency. For most workloads, you only need a baseline level of cloud management. For other workloads, you might spend two to four times more on cloud management compared to a baseline level. You can justify this cost because of the potential impact of business interruptions.

The previous articles in this series can help you understand the classification and impact of interruptions to various workloads. This article helps you calculate the returns.

The following diagram shows that each level of cloud management has inflection points in which cost can rise faster than resiliency. Those inflection points prompt detailed business decisions and business commitments.



Determine a proper commitment

For each workload in a portfolio, the cloud operations team and cloud strategy team should align on the level of management that the cloud operations team directly provides.

When your business establishes a commitment, determine how to align the following aspects.

- IT operations prerequisites
- Management responsibility
- Cloud tenancy
- Soft-cost factors
- Return on investment (ROI) loss avoidance
- Validation of the management level

To help you make decisions, the following sections describe these aspects in greater detail.

Determine IT operations prerequisites

The [Azure Management Guide](#) outlines Azure management tools. Before your business makes a commitment, IT should determine an acceptable standard-level management baseline to apply to all managed workloads. For each of the managed workloads in the IT portfolio, IT can then calculate a standard management cost that's based on CPU cores, disk space, and other asset-related variables. IT can also estimate a composite service-level objective (SLO) for each workload, based on the architecture.

IT operations teams often use a default minimum of 99.9% uptime for the initial composite SLO. They might normalize management costs based on the average workload, especially for solutions that have minimal logging and storage needs. To provide a starting point for initial conversations, the IT operations team can average the costs of a few medium-criticality workloads.

💡 Tip

If you use the [operations management workbook](#) to plan for cloud management, you should update the operations management fields to reflect the IT operations prerequisites. The operations management fields include *Commitment level*, *Composite SLO*, and *Monthly cost*. The monthly cost should represent the cost of the operational management tools that you add on a monthly basis.

The operations management baseline serves as an initial starting point, and you should also validate the baseline with the following aspects.

Choose a responsibility model

In a traditional on-premises environment, you might assume that the cost of managing the environment is a sunk cost for IT operations. A *sunk cost* is an expense that you can't recover. In the cloud, management is a purposeful decision that has a direct budgetary impact. You can directly attribute the costs of each management function to each workload that you deploy to the cloud. You have greater control with this approach. But the cloud operations teams and cloud strategy teams must first commit to an agreement about responsibilities.

Your business might also outsource some of your ongoing management functions to a [service provider](#). Service providers can use [Azure Lighthouse](#) to provide your business with precise control. For example, you can grant access to your resources and have greater visibility into the actions that service providers perform.

To manage your cloud environment, you can implement various models.

- **Delegated responsibility model:** IT operations can use an approach known as *delegated responsibility*. This approach doesn't require centralized management and prevents operational management overhead. In a cloud center of excellence (CCoE) model, platform operations and platform automation provide self-service management tools that business-led operations teams can use, independent of a centralized IT operations team.

This approach gives business stakeholders complete control over management-related budgets. The CCoE team can also ensure that a minimum set of guardrails is properly implemented. IT acts as a broker and a guide to help your business make wise decisions. Business operations oversee day-to-day operations of dependent workloads.

- **Centralized responsibility model:** Your business might require a *central IT team model* if you have compliance requirements, technical complexity, or some shared service models. In a central IT model, IT performs its operations management responsibilities.

You might centrally manage and control environmental design, management controls, and governance tooling, which prevents business stakeholders from making management commitments. But the visibility into the cost and the architecture of the cloud approaches makes it easier for centralized IT to communicate the cost and level of management for each workload.

- **Mixed model:** Classification is the foundation of a *mixed model* of management responsibilities. If your business is in the process of transforming from on-premises

to the cloud, you might require an on-premises-first operating model for some time. If your business has strict compliance requirements or depends on long-term contracts with IT outsourcing vendors, you might need a centralized operating model.

A mixed-model approach provides balance. In this approach, a central IT team provides a centralized operating model for all workloads that are mission critical or contain sensitive information. The team places all other workload classifications in a cloud environment that supports delegated responsibilities. The centralized responsibility approach serves as the general operating model, but your business has flexibility to adopt a specialized operating model based on your required level of support and sensitivity.

Consider who is responsible for the day-to-day operations management for a workload. Your responsibility approach affects your commitments.

Manage cloud tenancy

Typically, you can manage assets easier when they reside in a single tenant. But you might need to maintain multiple tenants. For more information about why you might require a multitenant Azure environment, see [Centralize management operations with Azure Lighthouse](#).

Consider soft-cost factors

The next section outlines an approach to determine comparative returns that are associated with various levels of management processes and tooling. For each analyzed workload, you can measure the cost of management relative to the forecasted impact of business disruptions. Use the following method to determine if you need to invest in more extensive management approaches.

Before you calculate the numbers, consider the soft-cost factors. Soft-cost factors produce a return, but that return is difficult to measure through direct hard-cost savings that are visible in a profit-and-loss statement. Soft-cost factors can indicate a need to invest in a higher level of management than is fiscally prudent.

A few examples of soft-cost factors include:

- Daily workload usage by the board or CEO.
- Workload usage by the top x% of customers that leads to a greater revenue impact elsewhere.

- Impact on employee satisfaction.

To make a commitment, the next data point that you should evaluate is a list of soft-cost factors. You don't need to document these factors at this stage, but make business stakeholders aware of their importance and their exclusion from the following calculations.

Calculate loss avoidance ROI

When the IT team that's responsible for cloud operations calculates the relative return on operations management costs, they should complete the previously mentioned prerequisites and assume a minimum level of management for all workloads.

The next commitment that your business should make is to accept the costs that are associated with the baseline-managed offering. Determine whether your business agrees to invest in the baseline offering to meet the minimum standards of cloud operations.

If your business doesn't agree to that level of management, you must create a solution so that your business can proceed. Ensure that your solution doesn't materially affect the cloud operations of other workloads.

You might want more than the standard management level. The following section helps validate that investment and the associated returns in the form of loss avoidance.

Increase levels of management

For managed solutions, you can apply several design principles and template solutions in addition to the management baseline. Each design principle for reliability and resiliency adds operating costs to the workload. IT and your business must agree on these extra commitments, so you must understand potential losses that you can avoid when you implement more principles.

The following calculations provide formulas to help you better understand the differences between losses and increased management investments. For more information about how to calculate the cost of increased management, see [Workload automation](#) and [Platform automation](#).

💡 Tip

If you use the [operations management workbook](#) to plan for cloud management, update the operations management fields to reflect each

conversation. These changes update the ROI formulas and each of the following fields.

Estimate outage

The composite SLO is the SLA that's based on the deployment of each asset in the workload. The composite SLO field drives the *estimated outage*, which is labeled `Est.` `outage` in the workbook. To calculate the estimated outage in hours per year without using the workbook, apply the following formula:

$$\text{Estimated outage} = (1 - \text{composite SLO percentage}) \times \text{number of hours in a year}$$

The workbook uses the default value of *8,760 hours per year*.

Standard loss impact

The *standard loss impact* forecasts the financial impact of any outage, assuming that the *estimated outage* prediction proves accurate. The standard loss impact is labeled `Standard Impact` in the workbook. To calculate this forecast without using the workbook, apply the following formula:

$$\text{Standard impact} = \text{estimated outage} @ \text{three 9s of uptime} \times \text{time-value impact}$$

The value serves as a baseline for cost if the business stakeholders invest in a higher level of management.

Composite-SLO impact

The *composite-SLO impact* provides the updated fiscal impact, based on the changes to the uptime SLA. Use this calculation to compare the projected financial impact of both options. The composite-SLO impact is labeled `Commitment level impact` in the workbook. To calculate this forecasted impact without the spreadsheet, apply the following formula:

$$\text{Composite-SLO impact} = \text{estimated outage} \times \text{time-value impact}$$

The value represents the potential losses that the changed commitment level and the new composite SLO should prevent.

Comparison basis

The *Comparison basis* field evaluates the standard impact and composite-SLO impact to determine the amount of return in the *Annual ROI* field.

Return on loss avoidance

If the cost of managing a workload exceeds the potential losses, the proposed investment in cloud management might not be worthwhile. To compare the *Return on loss avoidance*, see the column labeled *Annual ROI*. To calculate this column on your own, use the following formula:

$$\text{Return on loss avoidance} = (\text{comparison basis} - (\text{monthly cost} \times 12)) \div (\text{monthly cost} \times 12)$$

If you don't have other soft-cost factors to consider, you can use this comparison to quickly determine if you need to invest more in cloud operations, resiliency, reliability, or other areas.

Validate the commitment

At this point in the process, your business can make commitments, including centralized or delegated responsibility and Azure tenancy, and determine the level of commitment. You can validate and document each commitment to ensure that the cloud operations team, cloud strategy team, and business stakeholders align on these commitments to manage the workload.

Next step

After you make commitments, the responsible operations teams can configure the workload. To get started, evaluate various approaches to inventory and visibility.

[Inventory and visibility options](#)

Feedback

Was this page helpful?

 Yes

 No

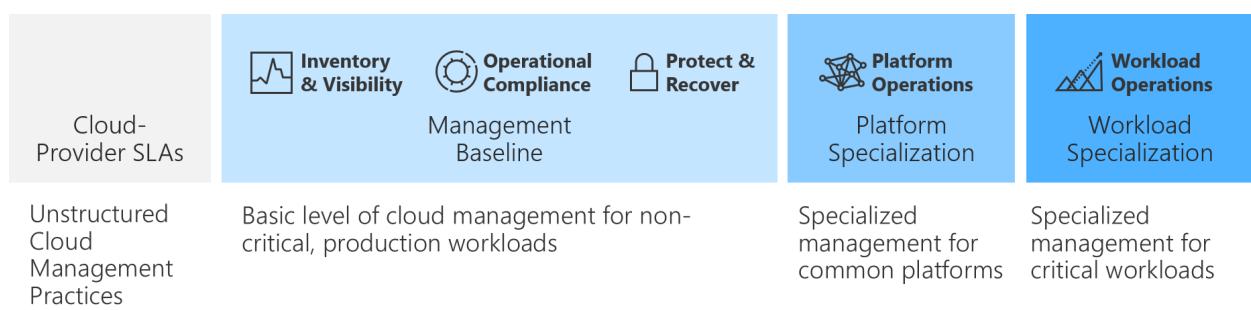
Cloud management disciplines

Article • 12/01/2022

The keys to proper management in any environment are consistency and repeatable processes. There are endless of options for the things that can be done in Azure. Likewise, there are countless approaches to cloud management. To provide consistency and repeatability, it's important to narrow those options to a consistent set of management processes and tools that will be offered for workloads hosted in the cloud.

Suggested management levels

Because the workloads in your IT portfolio vary, it's unlikely that a single level of management will suffice for each workload. To help you support a variety of workloads and business commitments, we suggest that your cloud operations team or platform operations team establish a few levels of operations management.



As a starting point, consider establishing the management levels that are shown in the preceding diagram and suggested in the following list:

- **Management baseline:** A cloud management baseline (or management baseline) is a defined set of tools, processes, and consistent pricing that serve as the foundation for all cloud management in Azure. To establish a cloud management baseline and determine which tools to include in the baseline offering to your business, review the list in the "cloud management disciplines" section.
- **Enhanced baseline:** Some workloads might require enhancements to the baseline that aren't necessarily specific to a single platform or workload. Although these enhancements aren't cost effective for every workload, there should be common processes, tools, and solutions for any workload that can justify the cost of the extra management support.
- **Platform specialization:** In any given environment, some common platforms are used by a variety of workloads. This general architectural commonality doesn't change when businesses adopt the cloud. Platform specialization is an elevated level of management that applies data and architectural subject matter expertise

to provide a higher level of operational management. Examples of platform specialization would include management functions specific to SQL Server, containers, Active Directory, or other services that can be better managed through consistent, repeatable processes, tools, and architectures.

- **Workload specialization:** For workloads that are truly mission critical, there might be a cost justification to go much deeper into the management of that workload. Workload specialization applies workload telemetry to determine more advanced approaches to daily management. That same data often identifies automation, deployment, and design improvements that would lead to greater stability, reliability, and resiliency beyond what's possible with operational management alone.

Organizations might also choose to [outsource functions related to one or more of these management levels to a service provider](#). These service providers can use [Azure Lighthouse](#) to provide greater precision and transparency.

The remaining articles in this series outline processes that are commonly found within these disciplines. In parallel, the [Azure Management Guide](#) demonstrates the tools that can support those processes. For assistance with building your management baseline, start with the Azure Management Guide. After you've established the baseline, this article series and the accompanying best practices can help expand that baseline to define other levels of management support.

Operations management discipline

Each suggested management level provides necessary operations management for all assets (applications, data, and infrastructure) in the portfolio, with an increasing degree of specificity. This mapping is designed to make it easier for the relevant roles to find the most appropriate processes and tools to deliver on the required level of cloud management.

The [operations management processes](#) defines the iterative discipline required to deliver operations and meet business commitments. This iterative process delivers on three levels of operations on a recurring rhythm of business:

- **Operations baseline (or enhanced baseline):** Consistent operations management of all deployed assets. Centralized technology teams focus on portfolio health and implements broad sweeping changes.
- **Platform operations:** Elevated operations for mission critical technology platforms. Centralized teams focus on operational fitness of shared platforms.
- **Workload operations:** Workload operations for refined changes to defined workloads. Workload specific teams deliver architecture change based on the

pillars of [Azure Well-Architected Framework](#) to improve operations through more granular improvements.

The [operations management processes](#) unite all three elevations of operations management in a holistic solution to improve operational fitness of all deployed assets regardless of the chosen operating model.

Next steps

The next step toward defining each level of cloud management is an understanding of [inventory and visibility](#).

[Inventory and visibility options](#)

Inventory and visibility in cloud management

Article • 04/04/2023

Operational management has a clear dependency on data. Consistent management requires an understanding about what is managed (inventory) and how those managed workloads and assets change over time (visibility). Clear insights about inventory and visibility help empower the team to manage the environment effectively. All other operational management activities and processes build on these two areas.

A few classic phrases about the importance of measurements set the tone for this article:

- Manage what matters.
- You can only manage what you can measure.
- If you can't measure it, it might not matter.

The inventory and visibility discipline builds on these timeless phrases. Before you can effectively establish operational management processes, it's important to gather data and create the right level of visibility for the right teams.

Common customer challenges

Unless inventory and visibility processes are consistently applied, operational management teams can suffer from a higher volume of business interruptions, longer time to recovery, and greater amounts of effort required to troubleshoot and triage issues. As changes adversely affect higher priority applications and larger numbers of assets, these metrics grow even faster.

These challenges stem from a few questions that can be answered only through consistent data and telemetry readings:

- How does the current-state performance deviate from standard operational performance telemetry?
- What assets are causing the business interruptions at the workload level?
- Which assets must be remediated to return to acceptable performance of this workload or business process?
- When did the deviation start? What was the trigger?
- Which changes have been made to the underlying assets? By whom?
- Were the changes intentional? Malicious?

- How did changes affect performance telemetry?

It's difficult, if not impossible, to answer these questions without a rich, centralized source for logs and telemetry data. To enable cloud management by ensuring the consistent configuration that's required to centralize the data, the baseline service must first start by defining the processes. Well-defined processes capture how a consistent configuration enforces data collection to support the components of inventory and visibility that are listed in the next section.

Components of inventory and visibility

Creating visibility on any cloud platform requires a few key components:

- Responsibility and visibility
- Inventory
- Central logging
- Change tracking
- Performance telemetry

Responsibility and visibility

When you establish commitments for each workload, [management responsibility](#) is a key factor. Delegated responsibility creates a need for delegated visibility. The first step toward inventory and visibility is to ensure that the responsible parties have access to the right data. Before you implement any cloud-native tools for visibility, ensure that each monitoring tool has the proper access and scope for each operations team.

Inventory

If no one knows that an asset exists, it's difficult to manage the asset. Before an asset or workload can be managed, it must be inventoried and classified. The first technical step toward stable operations is a validation of inventory and classification of inventory.

Central logging

Centralized logging is critical to the visibility that's required day to day by the operations management teams. We recommend that all assets that are deployed to the cloud record logs to a central location. In Azure, the central location is Log Analytics. Central logging drives reports about change management, service health, configuration, and most other aspects of IT operations.

Enforcing the consistent use of central logging is the first step toward establishing repeatable operations. Enforcement can be accomplished through corporate policy. When possible, however, you should automate enforcement to ensure consistency.

Change tracking

Change is the one constant in a technology environment. Awareness and understanding of changes across multiple workloads is essential to reliable operations. Your cloud management solution should include a means of understanding the 'when, how, and why' of technical change. Without those data points, remediation efforts are hindered.

Performance telemetry

Data drives business commitments about cloud management. To properly maintain commitments, the cloud operations team must understand the telemetry about the stability, performance, and operations of the workload. The cloud operations team must also understand the assets that support the workload.

The ongoing health and operations of the network, DNS, operating systems, and other foundational aspects of the environment are critical data points that factor into the overall health of any workload.

Processes

Compared to the features of the cloud management platform, the cloud management processes might be more important in your considerations, as they realize operations commitments with the business. Your cloud management methodology should include, at a minimum, the following processes:

- **Reactive monitoring:** Who addresses the deviations that adversely affect business operations? What actions do they take to remediate the deviations?
- **Proactive monitoring:** When deviations are detected but business operations aren't affected, how are those deviations addressed, and by whom?
- **Commitment reporting:** How is adherence to the business commitment communicated to business stakeholders?
- **Budgetary reviews:** What is the process for reviewing those commitments against budgeted costs? What is the process for adjusting the deployed solution or the commitments to create alignment?
- **Escalation paths:** What escalation paths are available when any of the preceding processes fail to meet the needs of the business?

There are several more processes related to inventory and visibility. The preceding list is designed to provoke thought within the operations team. Answering the list of questions helps to develop some of the necessary processes and might likely trigger new, deeper questions.

Responsibilities

When you're developing processes for operational monitoring, it's equally important to determine responsibilities for daily operation and regular support of each process.

- In a centralized IT organization, IT provides the operational expertise. The business is consultative in nature when issues require remediation.
- In a cloud center of excellence organization, business operations provide the expertise and hold responsibility for management of these processes. IT focuses on the automation and support of teams, as they operate the environment.

The preceding list items are examples of common responsibilities. Organizations often require a mixture of responsibilities to meet business commitments.

Act on inventory and visibility

Regardless of the cloud platform, the five components of inventory and visibility are used to drive most operational processes. All subsequent disciplines build on the data that's being captured. The next articles in this series outline ways to act on that data and integrate other data sources.

Share visibility

Data without action produces little return. Cloud management might expand beyond cloud-native tools and processes. To accommodate broader processes, a cloud management baseline might need to be enhanced to include reporting, IT service management integration, or data centralization. Cloud management might need to include one or more of the following principles during various phases of operational maturity.

Report

Offline processes and communication about commitments to business stakeholders often require reporting. Self-service reporting or periodic reporting might be a necessary component of an enhanced management baseline.

IT service management (ITSM) integration

ITSM integration is often the first example of acting on inventory and visibility. When deviations from expected performance patterns arise, ITSM integration uses alerts from the cloud platform to trigger tickets in a separate ITSM tool to trigger remediation activities. Some operating models might require ITSM integration as an aspect of the enhanced management baseline.

Data centralization

There's various reasons why a business might require multiple tenants within a single cloud provider. In those scenarios, data centralization is a required component of the enhanced management baseline, because it can provide visibility across those tenants or environments.

Next steps

Operational compliance builds on inventory capabilities by applying management automation and controls. See how [operational compliance](#) maps to your processes.

[Plan for operational compliance](#)

Operational compliance in cloud management

Article • 12/01/2022

Operational compliance builds on the discipline of [inventory and visibility](#). As the first actionable step of cloud management, this discipline focuses on regular telemetry reviews and remediation efforts (both proactive and reactive remediation). This discipline is the cornerstone for maintaining balance between security, governance, performance, and cost.

Components of operations compliance

Maintaining compliance with operational commitments requires analysis, automation, and human remediation. Effective operational compliance requires consistency in a few critical processes:

- Resource consistency
- Environment consistency
- Resource configuration consistency
- Update consistency
- Remediation automation

Resource consistency

The most effective step that a cloud management team can take toward operational compliance is to establish consistency in resource organization and tagging. When resources are consistently organized and tagged, all other operational tasks become easier. For deeper guidance on resource consistency, see the [Govern methodology](#). Specifically, review the [initial governance foundation articles](#) to learn how to start developing resource consistency.

Environment consistency

Establishing consistent environments, or landing zones, is the next most important step toward operational compliance. When landing zones are consistent and enforced through automated tools, it's significantly less complex to diagnose and resolve operational issues. For deeper guidance on environment consistency, see the [readiness phase](#) of the cloud adoption lifecycle. The exercises in this phase help build a repeatable

process for defining and maturing a consistent, code-first approach to developing cloud-based environments.

Resource configuration consistency

As it builds on governance and readiness approaches, cloud management should include processes for the ongoing monitoring and evaluation of its adherence to resource consistency requirements. As workloads change or new versions are adopted, it's vital that cloud management processes evaluate any configuration changes, which are not easily regulated through automation.

When inconsistencies are discovered, some are addressed by consistency in updates and others can be automatically remediated.

Update consistency

Stability in approach can lead to more stable operations. But some changes are required within cloud management processes. In particular, regular patching and performance changes are essential to reducing interruptions and controlling costs.

One of the many values of a mature cloud management methodology is a focus on stabilizing and controlling necessary change.

Any cloud management baseline should include a means of scheduling, controlling, and possibly automating necessary updates. Those updates should include patches at a minimum, but could also include performance, sizing, and other aspects of updating assets.

Remediation automation

As an enhanced baseline for cloud management, some workloads may benefit from automated remediation. When a workload commonly encounters issues that can't be resolved through code or architectural changes, automating remediation can help reduce the burden of cloud management and increase user satisfaction.

Many would argue that any issue that's common enough to automate should be resolved through resolution of technical debt. When a long-term resolution is prudent, it should be the default option. However, some business scenarios make it difficult to justify large investments in the resolution of technical debt. When such a resolution can't be justified, but remediation is a common and costly burden, automated remediation is the next best solution.

Next steps

[Protection and recovery](#) are the next areas to consider in a cloud management baseline.

[Protect and recover](#)

Protect and recover in cloud management

Article • 07/07/2023

Prior to preparing for a potential workload outage, cloud management teams should first make sure they've met requirements for:

- [inventory and visibility](#)
- [operational compliance](#)

As they plan, the teams must start with an assumption that something will fail when disaster strikes. Preparation for an outage allows the teams to detect failures sooner and recover more quickly. The focus of this discipline is on the steps that come immediately after a system fails. How do you protect workloads so that they can be recovered quickly when an outage occurs?

No technical solution can consistently offer an SLA that guarantees 100 percent uptime. Solutions with the most redundant architectures claim to deliver on "six 9s" or 99.9999 percent uptime. But even a "six 9s" solution goes down for 31.6 seconds in any given year. It's rare for a solution to warrant a large, ongoing operational investment that's required to reach "six 9s" of uptime.

Translate protection and recovery conversations

The workloads that power business operations consist of:

- applications
- data
- virtual machines (VMs)
- other assets

Each asset might require its own approach to protection and recovery. The important goal of this discipline is to establish a consistent commitment within the management baseline, which can provide a starting point for business discussions.

At a minimum, cloud management teams should create a baseline approach for each asset, with a clear commitment to quick recovery and minimal data loss.

Recovery time objectives (RTO)

A recovery time objective is the amount of time it should take to recover any system to its state prior to a disaster. This would include the time needed to:

- restore minimal functionality to VMs and applications
- restore data required by applications.

In business terms, RTO represents the amount of time that business processes are out of service. For mission-critical workloads, this variable should be relatively low, allowing business processes to resume quickly. For lower-priority workloads, a standard level of RTO might not have a noticeable impact on company performance.

A business should create a management baseline that establishes a standard RTO for non-mission-critical workloads. The business can then use that baseline as a way to justify additional investments in recovery times.

Recovery point objectives (RPO)

In most cloud management systems, some form of data protection periodically captures and stores data. The recovery point refers to the last time the data was captured. When a system fails, it can be restored only to the most recent recovery point.

The recovery point objective is measured from the most recent recovery point to an outage. If the RPO is measured in hours, a system failure results in the loss of data for the hours between the last recovery point and the outage. If the RPO is measured in days, a system failure results in the loss of data for the days between the last recovery point and the outage. A one-day RPO would theoretically result in the loss of all transactions in the day leading up to the failure.

For mission-critical systems, measuring an RPO in minutes or seconds might help avoid loss in revenue or profits. However, a shorter RPO generally results in increased management costs. To help minimize these costs, a business should create a management baseline that focuses on the longest acceptable RPO. The business can then decrease the RPO of the specific platforms or workloads that warrant more investment.

Protect and recover workloads

Most of the workloads in an IT environment support a specific business or technical process. Systems that don't have a systemic impact on business operations usually don't warrant the increased investment required to recover systems quickly or minimize data loss. By establishing a baseline, a business can figure out what level of recovery support

they need at a price point they can consistently manage. Understanding this helps business stakeholders evaluate the value of increased investment in recovery.

For most cloud management teams, an enhanced baseline, with specific RPO/RTO commitments for various assets, yields the most favorable path to mutual business commitments. The following sections outline a few common enhanced baselines that empower a business to easily add protection and recovery functionality through a repeatable process.

Protect and recover data

Data is arguably the most valuable asset in the digital economy. Loss of the data that powers a production workload leads to loss in revenue or profits. The most common enhanced baseline is the ability to protect and recover data effectively. We encourage cloud management teams to offer a level of enhanced management baseline that supports common data platforms.

Before cloud management teams implement platform operations, it's common for them to support improved operations for a platform as a service (PaaS) data platform. For instance, it's easy for a cloud management team to enforce a higher frequency of backup or multiregional replication for Azure SQL Database or Azure Cosmos DB solutions. Doing so allows the development team to easily improve RPO by modernizing their data platforms.

To learn more about this thought process, see [Platform operations discipline](#).

Protect and recover VMs

Most workloads somewhat depend on virtual machines, which host various aspects of the solution. A business must recover some virtual machines quickly for the workload to support its processes after a system failure.

Every minute of downtime on those virtual machines could cause lost revenue or reduced profits. When VM downtime has a direct impact on the fiscal performance of the business, RTO is very important. Cloud management teams can recover virtual machines quickly by replicating them to a secondary site and using automated recovery, a model that's referred to as a hot-warm recovery model. The teams can also replicate virtual machines to a functional, secondary site in an approach known as a hot-hot, or high-availability model. The hot-hot approach is more expensive, but it offers the highest state of recovery.

Each of these models reduces the RTO, which helps businesses restore their business capabilities faster. However, each model also results in significantly increased cloud management costs.

Also note that, apart from replication for high-availability, backup should be enabled for scenarios such as:

- accidental delete
- data corruption
- ransomware attacks

For more information about this thought process, see [Workload operations discipline](#).

Next steps

After this management baseline component is met, the team can look ahead to avoid outages in its [platform operations](#) and [workload operations](#).

[Platform operations](#)

[Workload operations](#)

Platform operations in cloud management

Article • 01/23/2023

A cloud management baseline that spans [inventory and visibility](#), [operational compliance](#), and [protection and recovery](#) might provide a sufficient level of cloud management for most workloads in the IT portfolio. But that baseline is seldom enough to support the full portfolio. This article builds on the most common next step in cloud management, portfolio operations.

A quick study of the assets in the IT portfolio highlights existing patterns across the supported workloads. Within those workloads, there are common platforms. The platforms could vary widely depending on the past technical decisions within the company.

For some organizations, there's a heavy dependence on SQL Server, Oracle, or other open-source data platforms. In other organizations, the commonalities might be rooted in the hosting platforms for virtual machines (VMs) or containers. Still others might have a common dependency on applications or enterprise resource planning (ERP) systems such as SAP or Oracle.

When you understand these commonalities, your cloud management team can specialize in higher levels of support for those prioritized platforms.

Establish a service catalog

The objective of platform operations is to create reliable and repeatable solutions for a cloud adoption team to use. The cloud adoption team can then deliver a platform that provides a higher level of business commitment. That commitment could decrease the likelihood or frequency of downtime, which improves reliability. If there's a system failure, the commitment can also help decrease the amount of data loss or time to recovery. Such a commitment often includes ongoing, centralized operations to support the platform.

As the cloud management team establishes higher degrees of operational management and specialization related to specific platforms, they add platforms to a growing service catalog. The service catalog provides self-service platform deployment in a specific configuration, which adheres to ongoing platform operations. During the business-alignment conversation, cloud management and cloud strategy teams can propose

service catalog solutions for the business. The service catalog solutions improve reliability, uptime, and recovery commitments in a controlled, repeatable process.

For reference, some organizations refer to an early-stage service catalog as an *approved list*. The primary difference is that a service catalog comes with ongoing operational commitments from the cloud center of excellence (CCoE). An approved list is similar. It provides a preapproved list of solutions that a team can use in the cloud. But typically there isn't an operational benefit associated with applications on an approved list.

Much like the debate between centralized IT and CCoE, the difference is one of priorities. A service catalog assumes good intent but provides operational, governance, and security guardrails that accelerate innovation. An approved list hinders innovation until operations, compliance, and security gates are passed for a solution. Both solutions are viable, but they require the company to make subtle prioritization decisions to invest more in innovation or compliance.

Build the service catalog

Cloud management is seldom successful at delivering a service catalog in a silo. Proper development of the catalog requires a partnership across the central IT team or the CCoE. This approach tends to be most successful when an IT organization reaches a CCoE level of maturity, but it could be implemented sooner.

When the cloud platform team builds the service catalog within a CCoE model, they build out the desired-state platform. The cloud governance and cloud security teams validate governance and compliance within the deployment. The cloud management team establishes ongoing operations for that platform. And the cloud automation team packages the platform for scalable, repeatable deployment.

After the platform is packaged, the cloud management team can add it to the growing service catalog. From there, the cloud adoption team uses the package or others in the catalog during deployment. After the solution goes to production, the business realizes the extra benefits of improved operational management and potentially reduced business disruptions.

Note

Building a service catalog requires a great deal of effort and time from multiple teams. Using the service catalog or approved list as a gating mechanism slows innovation. When innovation is a priority, develop service catalogs in parallel to other adoption efforts.

Define your own platform operations

Although management tools and processes can improve platform operations, it's often not enough to achieve the desired states of stability and reliability. True platform operations require a focus on pillars of architecture excellence. When a platform justifies a deeper investment in operations, consider the following five pillars before the platform becomes a part of any service catalog:

- **Reliability:** Design systems to recover from failures and continue to function.
- **Security:** Protect applications and data from threats.
- **Cost optimization:** Manage costs to maximize the value delivered.
- **Operational excellence:** Follow operational processes that keep a system running in production.
- **Performance efficiency:** Scale systems to adapt to changes in load.

The [Microsoft Azure Well-Architected Framework](#) provides an approach to evaluating specific workloads for adherence to these pillars, to improve overall operations. You can apply these pillars to both platform operations and workload operations.

Get started with specific platforms

The platforms discussed in the next sections are common to typical Azure customers, and they can easily justify an investment in platform operations. Cloud management teams tend to start with them when they're building out platform operations requirements or a full service catalog.

PaaS data operations

Data is often the first platform to warrant platform operations investments. When data is hosted in a platform as a service (PaaS) environment, business stakeholders tend to request a reduced recovery point objective (RPO) to minimize data loss. Depending on the nature of the application, they might also request a reduction in recovery time objective (RTO). In either case, the architecture that supports PaaS-based data solutions can easily accommodate some increased level of management support.

In most scenarios, the cost of improving management commitments is easily justified, even for applications that aren't mission critical. This platform operations improvement is so common that many cloud management teams see it more as an enhanced baseline, rather than as a true platform operations improvement.

IaaS data operations

When data is hosted in a traditional infrastructure as a service (IaaS) solution, the effort to improve RPO and RTO can be higher. Yet the business stakeholders' desire to achieve better management commitments is seldom affected by a PaaS versus IaaS decision. If anything, an understanding of the fundamental differences in architecture might prompt the business to ask for PaaS solutions or commitments that match what's available on PaaS solutions. Consider modernization of any IaaS data platforms as a first step into platform operations.

When modernization isn't an option, cloud management teams commonly prioritize IaaS-based data platforms as a first required service in the service catalog. Providing the business with a choice between standalone data servers and clustered, high-availability, data solutions makes the business commitment conversation much easier to facilitate. A basic understanding of the operational improvements and increased costs helps the business make the best decision for its business processes and supporting workloads.

Other common platform operations

In addition to data platforms, virtual machine hosts tend to be a common platform for operations improvements. Cloud platform and cloud management teams most commonly invest in improvements to VMware hosts or container solutions. Such investments can improve the stability and reliability of the hosts, which support the VMs, which in turn power the workloads. Proper operations on one host or container can improve the RPO or RTO of several workloads. This approach creates improved business commitments but distributes the investment. Improved commitments and reduced costs combine to make it much easier to justify improvements to cloud management and platform operations.

Next steps

In parallel with improvements to platform operations, cloud management teams also focus on improving [workload operations](#) for the top 20 percent or less of production workloads.

[Improve workload operations](#)

Workload operations in cloud management

Article • 12/01/2022

Some workloads are critical to the success of the business. For those workloads, a management baseline is insufficient to meet the required business commitments to cloud management. Platform operations might not even be sufficient to meet business commitments. This highly important subset of workloads requires a specialized focus on the way the workload functions and how it's supported.

In return, the investment in workload operations can lead to improved performance, decreased risk of business interruption, and faster recovery when system failures occur. This article discusses an approach to investing in the continued operations of these high priority workloads to drive improved business commitments.

When to invest in workload operations

The *Pareto principle* (also known as the *80/20 rule*) states that 80 percent of effects come from 20 percent of the causes. When IT portfolios are allowed to grow organically over time, this rule is often illustrated in a review of the IT portfolio. Depending on the effect that requires investment, the cause can vary but the general principle holds true:

- 80 percent of system failures tend to be the result of 20 percent of the common errors or bugs.
- 80 percent of business value tends to come from 20 percent of the workloads in a portfolio.
- 80 percent of the effort to migrate to the cloud comes from 20 percent of the workloads being moved.
- 80 percent of cloud management efforts will support 20 percent of the service incidents or trouble tickets.
- 80 percent of business impact from an outage will come from 20 percent of the systems affected by the outage.

Workload operations should be applied only when the cloud adoption strategy, business outcomes, and operational metrics are each well understood. This is a paradigm shift from the classic view of IT. Traditionally, IT assumed that all workloads experienced the same degree of support and required similar levels of priority.

Before they invest in deep workload operations, both IT and the business should understand the business justifications and the expectations of increased investment in

cloud management.

Start with the data

Workload operations begin with a deep understanding of workload performance and support requirements. Before the team invests in workload operations, it must have rich data about workload dependencies, application performance, database diagnostics, virtual machine telemetry, and incident history.

This data seeds the insights that drive workload operations decisions.

Continued observation

Initial data and ongoing telemetry can help formulate and test theories about the performance of a workload. But ongoing workload operations are rooted in a continued and expanded observation of workload performance, with a heavy focus on application and data performance.

Test the automation

At the application level, the first requirements of workload operations, is an investment in deep testing. For any application that's supported through workload operations, a test plan should be established and regularly executed to deliver functional and scale testing across the applications.

Regular test telemetry can provide immediate validation of various hypotheses about the operation of the workload. Improving operational and architectural patterns can be executed and tested. The resulting deltas provide a clear impact analysis to guide continued investments.

Understand releases

A clear understanding of release cycles and release pipelines is an important element of workload operations.

An understanding of cycles can prepare for potential interruptions and allow the team to proactively address any releases that might produce an adverse effect on operations. This understanding also allows the cloud management team to partner with adoption teams to continuously improve the quality of the product and address any bugs that might affect stability.

More importantly, an understanding of release pipelines can significantly improve the recovery point objective (RPO) of a workload. In many scenarios, the fastest and most accurate path to the recovery of an application is a release pipeline. For application layers that change only when a new release happens, it might be wise to invest more heavily in pipeline optimization than on the recovery of the application from traditional back-up processes.

Although a deployment pipeline can be the fastest path to recovery, it can also be the fastest path to remediation. When an application has a fast, efficient, and reliable release pipeline, the cloud management team has an option to automate deployment to a new host as a form of automated remediation.

There might be many other faster, more effective mechanisms for remediation and recovery. However, when the use of an existing pipeline can meet business commitments and capitalize on existing DevOps investments, the existing pipeline might be a viable alternative.

Clearly communicate changes to the workload

Change to any workload is among the biggest risks to workload operations. For any workload in the workload operations level of cloud management, the cloud management team should closely align with the cloud adoption teams to understand the changes coming from each release. This investment in proactive understanding will have a direct, positive impact on operational stability.

Improve outcomes

The data and communication investments in a workload will yield suggestions for improvements to ongoing operations in one of three areas:

- Technical debt resolution
- Automated remediation
- Improved system design

Technical debt resolution

The best workload operations plans still require remediation. As your cloud management team seeks to stay connected to understand adoption efforts and releases, the team likewise should regularly share remediation requirements to ensure that technical debt and bugs are a continued priority for your development teams.

Automated remediation

By applying the Pareto principle, we can say that 80 percent of negative business impact likely comes from 20 percent of the service incidents. When those incidents can't be addressed in normal development cycles, investments in remediation automation can significantly reduce business interruptions.

Improved system design

In the cases of technical debt resolution and automated remediation, system flaws are the common cause of most system outages. You can have the greatest impact on overall workload operations by adhering to a few design principles:

- **Scalability:** The ability of a system to handle increased load.
- **Availability:** The percentage of time that a system is functional and working.
- **Resiliency:** The ability of a system to recover from failures and continue to function.
- **Management:** Operations processes that keep a system running in production.
- **Security:** Protecting applications and data from threats.

To help improve overall operations, the [Microsoft Azure Well-Architected Framework](#) provides an approach to evaluating specific workloads for adherence to these pillars. Apply the pillars to both platform operations and workload operations.

Next steps

With a full understanding of the Manage methodology within the Cloud Adoption Framework, you are now armed to implement cloud management principles. Learn how to make this methodology actionable within your operations environment.

[Apply this methodology](#)

Sustainability alignment in cloud management

Article • 06/18/2024

Introducing sustainability in your cloud management and operations help drive carbon awareness in your teams and ultimately work toward achieving your goals set up for your [sustainability outcomes](#).

Monitoring carbon emissions

Utilize monitoring capabilities to understand better how your organization uses resources and help identify areas of improvement.

In the Azure Well-Architected Framework, we describe measuring and tracking carbon impact using the Emissions Impact Dashboard and Azure carbon optimization, defining emissions targets, identifying the metrics and setting improvement goals, using cost optimization as a proxy for carbon, and defining policies. To learn more, see [operational procedure considerations for sustainable workloads on Azure](#).

Cost as a proxy for sustainability

While sustainability isn't usually the primary purpose of [reducing service costs](#) with tools like Azure Advisor, these tools can often be aligned with carbon savings. Consider [cost as a proxy for sustainability](#), and see how an optimized workload becomes leaner and ultimately reduces the carbon footprint.

Find opportunities to schedule workloads

Part of your continuous operations and management of the cloud estate should be to evaluate what workloads you can schedule. For example, [running batch workloads during low-carbon periods](#).

Monitor for services to retire

Understanding what services you're actively using and monitoring for unused resources can help you iteratively increase your cloud efficiency, lowering your carbon footprint.

Continuously managing your cloud estate is essential, including understanding what portions of your Azure resources aren't being used. An easy way to operationalize this is

by [using PowerShell to identify unassociated resources in Azure](#).

Remove unused data

Improve the sustainability of your IT operations by moving data to tape or a long-term archive. This offers insights into the "invisible sinks" within an organization. We can then also shift to quality from quantity.

To dive deeper, see [Learn How Moving Data to Tape Can Lead to Significant Energy Savings and Reduction in CO2 Emissions](#).

Sustainability insights

When reporting on sustainability and cloud management, it's essential to consider internal and external stakeholders. Internally, you can share reports with relevant teams and management to raise awareness and promote accountability. Externally, customers, investors, or other stakeholders might request reports to evaluate the organization's sustainability and environmental impact.

By tracking and reporting on key metrics, organizations can promote transparency, accountability, and continuous sustainability and environmental impact improvement.

Emissions Impact Dashboard

In Azure, customers can use the [Emissions Impact Dashboard](#) to track and get insights on carbon emissions. This tool provides valuable insights into emissions by subscription, region, and service, allowing customers to understand their environmental impact better.

Customers can access information on emissions scopes, years, months, and other details, providing a comprehensive view of their emissions. This centralized tool can be valuable for large enterprises with complex cloud environments in tracking emissions accurately and efficiently.

Azure carbon optimization

Use [Azure carbon optimization](#) to measure and minimize the carbon impact of your Azure resources. With Carbon optimization, you can find opportunities to optimize resource utilization to lower carbon emissions and costs, track and analyze emissions associated with Azure resources and subscriptions, and access carbon data and insights

through APIs and exports. Carbon optimization provides emission data for all Azure resource types, based on billing and usage.

Demand shaping

Demand shaping is a technique to optimize resource utilization by aligning demand with available resources. From a sustainability perspective, demand shaping can effectively reduce carbon emissions by ensuring that resources are used efficiently and effectively.

In a cloud environment, demand shaping typically involves identifying periods of low demand and [scheduling resource-intensive workloads during those times](#). By doing so, organizations can take advantage of excess capacity and reduce the need to deploy more resources, which can help reduce energy consumption and carbon emissions.

From a management perspective, demand shaping requires careful planning and coordination. It's essential to have a clear understanding of resource utilization patterns and workload requirements and the ability to automate resource allocation and scheduling.

Feedback

Was this page helpful?



Apply design principles and advanced operations

Article • 12/01/2022

The first three cloud management disciplines describe a management baseline. At a minimum, a management baseline should include a standard business commitment to minimize business interruptions and accelerate recovery if service is interrupted. Most management baselines include a disciplined focus on maintaining *inventory and visibility, operational compliance, and protection and recovery*.

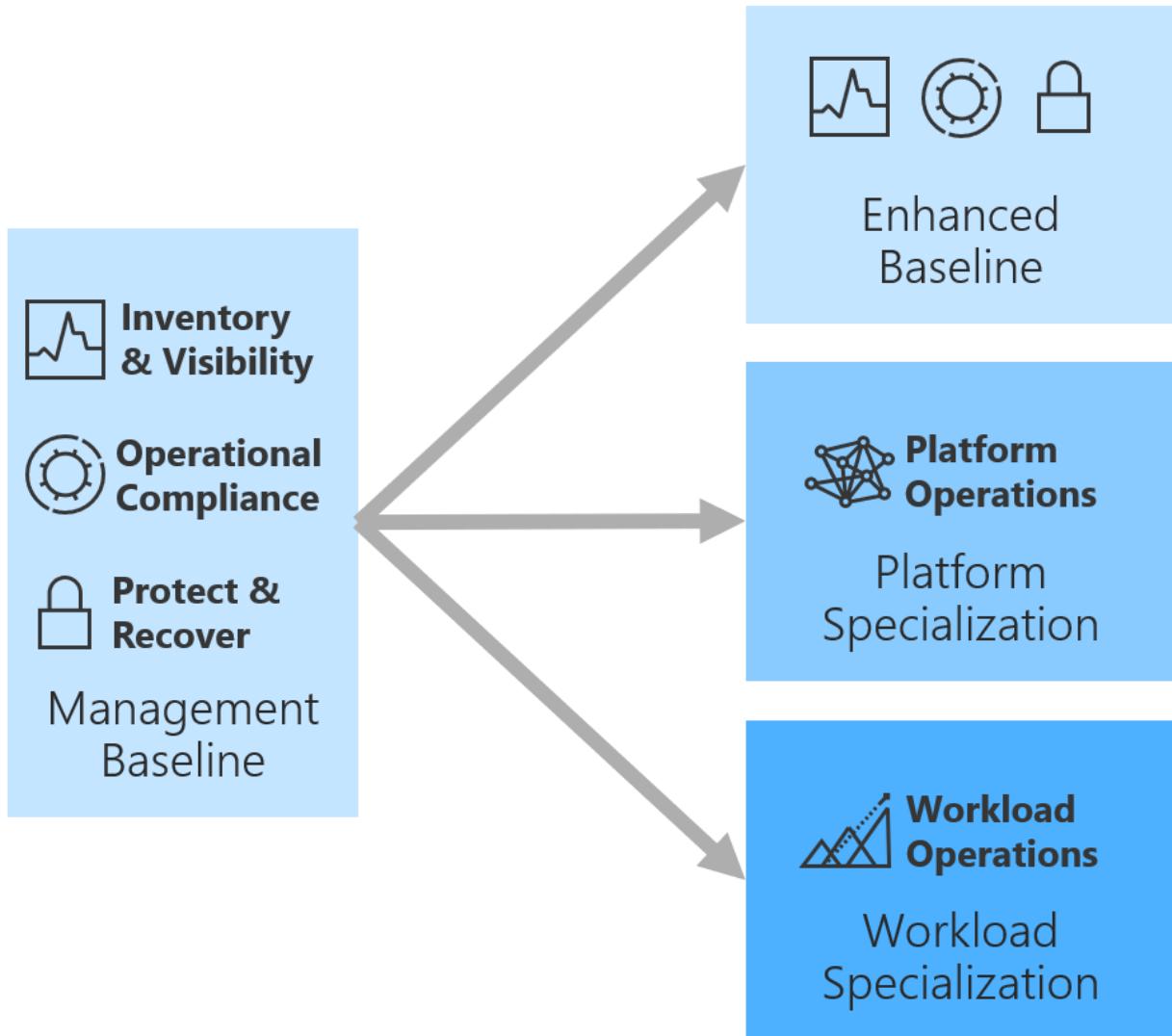
The purpose of a management baseline is to create a consistent offering that provides a minimum level of business commitment for all supported workloads. This baseline of common, repeatable management offerings allows the team to deliver a highly optimized degree of operational management, with minimal deviation. But that standard offering might not provide a rich enough commitment to the business.

The diagram in the next section illustrates three ways to go beyond the management baseline.

The management baseline should meet the minimum commitment required by 80 percent of the lowest criticality workloads in the portfolio. The baseline should not be applied to mission-critical workloads. Nor should it be applied to common platforms that are shared across workloads. Those workloads require a focus on design principles and advanced operations.

Advanced operations options

There are three suggested paths for improving business commitments beyond the management baseline, as shown in the following diagram:



Enhanced management baseline

As outlined in the Azure Management Guide, an enhanced management baseline uses cloud-native tools to improve uptime and decrease recovery times. The improvements are significant, but less so than with workload or platform specialization. The advantage of an enhanced management baseline is the equally significant reduction in cost and implementation time.

Management specialization

Aspects of workload and platform operations might require changes to design and architecture principles. Those changes could take time and might result in increased operating expenses. To reduce the number of workloads requiring such investments, an enhanced management baseline could provide enough of an improvement to the business commitment.

For workloads that warrant a higher investment to meet a business commitment, specialization of operations is key.

Areas of management specialization

There are two areas of specialization:

- **Platform specialization:** Invest in ongoing operations of a shared platform, distributing the investment across multiple workloads.
- **Workload specialization:** Invest in ongoing operations of a specific workload, generally reserved for mission-critical workloads.

Central IT team or cloud center of excellence (CCoE)

Decisions between platform specialization and workload specialization are based on the criticality and impact of each workload. However, these decisions are also indicative of larger cultural decisions between central IT team and CCoE organizational models.

Workload specialization often triggers a cultural change. Traditional IT and centralized IT both build processes that can provide support at scale. Scale support is more achievable for repeatable services found in a management baseline, enhanced baseline, or even platform operations. Workload specialization doesn't often scale. This lack of scale makes it difficult for a centralized IT organization to provide necessary support without reaching organizational scale limitations.

Alternatively, a cloud center of excellence approach scales through purposeful delegation of responsibility and selective centralization. Workload specialization tends to better align with the delegated responsibility approach of a CCoE.

The natural alignment of roles in a CCoE is outlined as follows:

- The cloud platform team helps build common platforms that support multiple cloud adoption teams.
- The cloud automation team extends those platforms into deployable assets in a service catalog.
- Cloud management delivers the management baseline centrally and helps support the use of the service catalog.
- But the business unit (in the form of a business DevOps team or cloud adoption team) holds responsibility for day-to-day operations of the workload, pipeline, or performance.

As for aligning areas of management, central IT team and CCoE models can generally deliver on platform specialization, with minimal cultural change. Delivering on workload specialization might be more complex for central IT teams.

Management specialization processes

Within each specialization, the following four-step process is delivered in a disciplined, iterative approach. This approach requires partnership among cloud adoption, cloud platform, cloud automation, and cloud management experts to create a viable and informed feedback loop.

- **Improve system design:** Improve the design of common systems (platforms) or specific workloads to effectively minimize interruptions.
- **Automate remediation:** Some improvements are not cost effective. In such cases, it might make more sense to automate remediation and reduce the impact of interruptions.
- **Scale the solution:** As systems design and automated remediation are improved, you can scale those changes across the environment through the service catalog.
- **Continuous improvement:** You can use various monitoring tools to discover incremental improvements to address in the next pass of system design, automation, and scale.

Improve system design

Improving system design is the most effective approach to improving operations of any common platform. System design improvements can help increase stability and decrease business interruptions. Design of individual systems is out of scope for the environment view taken throughout the Cloud Adoption Framework.

As a complement to this framework, the [Microsoft Azure Well-Architected Framework](#) provides guiding tenets for improving the quality of a platform or a specific workload. The framework focuses on improvement across five pillars of architecture excellence:

- **Cost optimization:** Manage costs to maximize the value delivered.
- **Operational excellence:** Follow operational processes that keep a system running in production.
- **Performance efficiency:** Scale systems to adapt to changes in load.
- **Reliability:** Design systems to recover from failures and continue to function.
- **Security:** Protect applications and data from threats.

Most business interruptions equate to some form of technical debt, or deficiency in the architecture. For existing deployments, systems design improvements can be viewed as

payments against existing technical debt. For new deployments, systems design improvements can be viewed as avoidance of technical debt. The next section shows how to deal with technical debt that can't or shouldn't be addressed.

To improve system design, learn more about the [Microsoft Azure Well-Architected Framework](#). As your system design improves, return to this article to find new opportunities to improve and scale the improvements across your environment.

Automated remediation

Some technical debt can't or shouldn't be addressed. Resolution could be too expensive to correct. It could be planned but might have a long project duration. The business interruption might not have a significant business impact, or the business priority is to recover quickly instead of investing in resiliency.

When resolution of technical debt isn't the desired path, automated remediation is commonly the desired next step. Using Azure Automation and Azure Monitor to detect trends and provide automated remediation is the most common approach to automated remediation.

For guidance on automated remediation, see [Azure Automation and alerts](#).

Scale the solution with a service catalog

The cornerstone of platform specialization and platform operations is a well-managed service catalog. This is how improvements to systems design and remediation are scaled across an environment. The cloud platform team and cloud automation team align to create repeatable solutions to the most common platforms in any environment. However, if those solutions aren't consistently applied, cloud management can provide little more than a baseline offering.

To maximize adoption and minimize maintenance overhead of any optimized platform, the platform should be added to a service catalog. Each application in the catalog can be deployed for internal consumption via the service catalog, or as a marketplace offering for external consumers.

For information about publishing to a service catalog, see the series on [publishing to a service catalog](#).

Continuous improvement

Platform specialization and platform operations both depend on strong feedback loops between adoption, platform, automation, and management teams. Grounding those feedback loops in data empowers each team to make wise decisions. For platform operations to achieve long-term business commitments, it's important to take advantage of insights that are specific to the centralized platform. Because containers and SQL Server are the two most common centrally managed platforms, consider beginning with continuous improvement data collection by reviewing the following articles:

- [Container performance](#)
- [PaaS database performance](#)
- [IaaS database performance](#)

Cloud operation and management antipatterns

Article • 03/22/2023

Customers often experience antipatterns during the operation or management phase of cloud adoption. By introducing new or modernized tool chains with caution, you can often avoid these antipatterns.

Antipattern: Focus on tooling, not business outcomes

Modernized IT tooling can improve work environments by relieving employees of tedious tasks. It's important to measure new IT tooling so that you can identify whether it improves business outcomes. A new or modernized tool chain doesn't automatically provide faster delivery or a better business outcome.

Example: Introduce a platform that doesn't improve performance

A company introduces a new, improved version of its continuous integration and continuous delivery (CI/CD) platform. The tool makes it easier to define delivery and deployment pipelines, so you can deploy features faster. The IT department is enthusiastic about delivering a platform that speeds up the pipeline configuration. Once a business unit uses the tool, it discovers that the time to market isn't significantly better, compared with the old platform. The final approval and release process isn't changed or improved.

Preferred outcome: Measure success with business outcomes

To keep your technology and business goals aligned, have leaders from both areas jointly define desired outcomes. Make sure these outcomes and goals are specific, measurable, achievable, reasonable, and time-bound (SMART). Ensure that the outcomes and goals have an impact on technology and the business. The Microsoft Cloud Adoption Framework for Azure can help to [determine a proper commitment within the business](#).

Don't use simple technology outputs (such as faster deployment and pipeline configurations) to measure success. Instead, use technology and business outcomes. For help with this task, see [Developer velocity](#).

Next steps

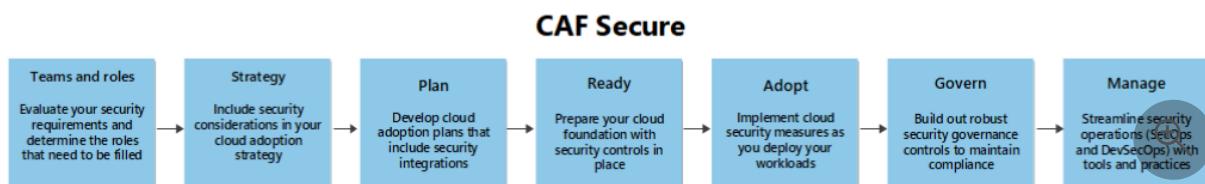
- [Business commitment in cloud management](#)

Secure overview

Article • 11/18/2024

The Cloud Adoption Framework for Azure Secure methodology provides a structured approach for securing your Azure cloud estate.

The guidance in this series of articles provides recommendations relevant to all methodologies within the Cloud Adoption Framework because security should be an integral part of every phase of your cloud adoption journey. Therefore, you can find articles aligned with each methodology that provide security recommendations for you to consider as you progress through each phase of your cloud adoption journey.



All of the recommendations in this guidance adhere to the [Zero Trust principles](#) of assume compromise (or assume breach), least privilege, and explicit verification of trust that should guide your security strategy, architecture, and implementation.

Holistic security guidance

Security is a complex and challenging discipline that you need to consider in nearly all aspects of your cloud and technology environments. Consider the following key points:

- **Anything is a potential target or attack vector:** In today's world, attackers can exploit any weaknesses in an organization's people, processes, and technologies to accomplish their malicious goals.
- **Security is a team sport:** To defend against these attacks, a coordinated approach is required across business, technology, and security teams. Each team must contribute to security efforts and collaborate effectively. For information about the various roles required to secure Azure resources, see [Teams and roles](#).

This Cloud Adoption Framework Secure guidance is one component of a larger holistic set of Microsoft security guidance designed to help various teams understand and perform their security responsibilities. The complete set includes the following guidance:

- **The Cloud Adoption Framework Secure methodology** provides security guidance for teams that manage the technology infrastructure that supports all the workload

development and operations hosted on Azure.

- [Azure Well-Architected Framework security guidance](#) provides guidance for individual workload owners about how to apply security best practices to application development and DevOps and DevSecOps processes. Microsoft provides guidance that complements this documentation about how to apply [security practices](#) and DevSecOps controls in a security development lifecycle.
- [Microsoft Cloud Security Benchmark](#) provides best practice guidance for stakeholders to ensure robust cloud security. This guidance includes security baselines that describe the available security features and recommended optimal configurations for Azure services.
- [Zero Trust guidance](#) provides guidance for security teams to implement technical capabilities to support a Zero Trust modernization initiative.

Each article covers several topics related to its aligned methodology:

- Security posture modernization
- Incident preparation and response
- The Confidentiality, Integrity, and Availability (CIA) Triad
- Security posture sustainment

Security posture modernization

Throughout your cloud adoption journey, look for opportunities to enhance your overall security posture through modernization. The guidance in this methodology is aligned with the [Microsoft Zero Trust adoption framework](#). This framework provides a detailed, step-by-step approach to modernizing your security posture. As you review the recommendations for each phase of the Cloud Adoption Framework methodology, enhance them by using the guidance provided in the Zero Trust adoption framework.

Incident preparation and response

Incident preparation and response are cornerstone elements of your overall security posture. Your ability to prepare for and respond to incidents can significantly affect your success in operating within the cloud. Well-designed preparation mechanisms and operational practices enable quicker threat detection and help minimize the blast radius of incidents. This approach facilitates faster recovery. Similarly, well-structured response mechanisms and operational practices ensure efficient navigation through recovery activities and provide clear opportunities for continuous improvement throughout the

process. By focusing on these elements, you can enhance your overall security strategy, which ensures resilience and operational continuity in the cloud.

The CIA Triad

The CIA Triad is a fundamental model in information security that represents three core principles. These principles are confidentiality, integrity, and availability.

- **Confidentiality** ensures that only authorized individuals can access sensitive information. This policy includes measures like encryption and access controls to protect data from unauthorized access.
- **Integrity** maintains the accuracy and completeness of data. This principle means protecting data from alterations or tampering by unauthorized users, which ensures that the information remains reliable.
- **Availability** ensures that information and resources are accessible to authorized users when needed. This task includes maintaining systems and networks to prevent downtime and ensure continuous access to data.

Adopt the CIA Triad to ensure that your business technology remains reliable and secure. Use it to enforce reliability and security in your operations through well-defined, strictly followed, and proven practices. Some ways that the triad principles can help ensure security and reliability are:

- **Data protection:** Protect sensitive data from breaches by taking advantage of the CIA Triad, which ensures privacy and compliance with regulations.
- **Business continuity:** Ensure data integrity and availability to maintain business operations and avoid downtime.
- **Customer trust:** Implement the CIA Triad to build trust with customers and stakeholders by demonstrating a commitment to data security.

Each methodology-aligned article provides recommendations for the principles of the CIA Triad. This approach ensures that you can address confidentiality, integrity, and availability. This guidance helps you thoroughly consider these aspects in every phase of your cloud adoption journey.

Security posture sustainment

Continuous improvement is crucial for maintaining a robust security posture in the cloud because cyber threats continuously evolve and become more sophisticated. To

protect against these ever-changing risks, ensure ongoing enhancements. The guidance in these sections can help you set up your organization for long-term success by identifying opportunities for continuous improvement. Focus on these strategies as you establish and evolve your cloud environment over time.

Cloud security checklist

Use the cloud security checklist to see all tasks for each cloud security step. Quickly navigate to the guidance that you need.

[] Expand table

Cloud security step	Cloud security tasks
<input type="checkbox"/> Understand security teams and roles.	<input type="checkbox"/> Understand the role of the cloud service provider. <input type="checkbox"/> Understand the roles of Infrastructure and Platform teams. <input type="checkbox"/> Understand the roles of Security architecture, engineering, posture management teams. <input type="checkbox"/> Understand the roles of the Security Operations (SecOps and SOC) teams. <input type="checkbox"/> Understand the roles of Security Governance, Risk, and Compliance (GRC) teams. <input type="checkbox"/> Learn about security education and policy.
<input type="checkbox"/> Integrate security into your cloud adoption strategy.	<input type="checkbox"/> Security posture modernization strategy. <input type="checkbox"/> Incident preparedness and response strategy. <input type="checkbox"/> Confidentiality strategy. <input type="checkbox"/> Integrity strategy. <input type="checkbox"/> Availability strategy. <input type="checkbox"/> Security posture sustainment strategy
<input type="checkbox"/> Plan for a secure cloud adoption.	<input type="checkbox"/> Plan for landing zone adoption. <input type="checkbox"/> Security posture modernization planning. <input type="checkbox"/> Incident preparedness and response planning. <input type="checkbox"/> Confidentiality planning. <input type="checkbox"/> Integrity planning <input type="checkbox"/> Availability planning <input type="checkbox"/> Security posture sustainment planning
<input type="checkbox"/> Ready your secure cloud estate.	<input type="checkbox"/> Ready for security posture modernization. <input type="checkbox"/> Ready for incident preparedness and response. <input type="checkbox"/> Ready for confidentiality. <input type="checkbox"/> Ready for integrity. <input type="checkbox"/> Ready for availability <input type="checkbox"/> Ready for security posture sustainment

Cloud security step	Cloud security tasks
<input type="checkbox"/> Perform your cloud adoption securely.	<input type="checkbox"/> Security posture modernization adoption. <input type="checkbox"/> Adopt incident preparedness and response. <input type="checkbox"/> Adopt confidentiality. <input type="checkbox"/> Adopt integrity. <input type="checkbox"/> Adopt availability. <input type="checkbox"/> Adopt security posture sustainment
<input type="checkbox"/> Securely govern your cloud estate.	<input type="checkbox"/> Security posture modernization. <input type="checkbox"/> Incident preparedness and response governance <input type="checkbox"/> Confidentiality governance. <input type="checkbox"/> Integrity governance. <input type="checkbox"/> Availability governance. <input type="checkbox"/> Sustaining security governance
<input type="checkbox"/> Securely manage your cloud estate.	<input type="checkbox"/> Security posture modernization. <input type="checkbox"/> Managing incident preparedness and response <input type="checkbox"/> Managing confidentiality. <input type="checkbox"/> Managing integrity. <input type="checkbox"/> Managing availability. <input type="checkbox"/> Managing security sustainment

Next step

Security teams, roles, and functions

Feedback

Was this page helpful?

 Yes

 No

Security teams, roles, and functions

Article • 11/18/2024

This article describes the security roles required for cloud security and the functions they perform related to cloud infrastructure and platforms. These roles help you ensure that security is part of every stage of the cloud lifecycle, from development to operations and continuous improvement.



ⓘ Note

The Cloud Adoption Framework for Azure focuses on cloud infrastructure and platforms that support multiple workloads. For security guidance for individual workloads, see the [security guidance](#) in the Azure Well-Architected Framework.

Depending on your organization's size and other factors, the roles and functions discussed in this article might be fulfilled by people who perform multiple functions (roles) rather than by a single person or team. Enterprises and large organizations tend to have larger teams with more specialized roles, whereas smaller organizations tend to consolidate multiple roles and functions among a smaller number of people. The specific security responsibilities also vary depending on the technical platforms and services that the organization uses.

Some security tasks will be performed directly by technology and cloud teams. Others might be performed by specialized security teams that operate collaboratively with the technology teams. Regardless of the size and structure of your organization, stakeholders must have a clear understanding of the security jobs that need to be done. Everyone must also be aware of the business requirements and the security risk tolerance of the organization so they can make good decisions about cloud services that take into account and balance security as a key requirement.

Use the guidance in this article to help understand specific functions that teams and roles perform and how different teams interact to cover the entirety of a cloud security organization.

Transformation of security roles

Security architecture, engineering, and operations roles are undergoing a significant transformation of their responsibilities and processes. (This transformation is similar to the cloud-driven transformation of infrastructure and platform roles.) This security role transformation has been driven by multiple factors:

- As security tools increasingly become SaaS based, there's less need to design, implement, test, and operate security tool infrastructures. These roles do still need to support the full lifecycle of configuring cloud services and solutions (including continuous improvement) to ensure they meet security requirements.
- The recognition that security is everyone's job is driving a more collaborative and mature approach that enables security and technology teams to work together:
 - Technical engineering teams are accountable for ensuring that security measures are applied effectively to their workloads. This change increases their need for context and expertise from security teams on how to meet these obligations effectively and efficiently.
 - Security teams are shifting from a (slightly adversarial) quality control role to a role that enables technical teams: making the secure path the easiest path. Security teams reduce friction and barriers by using automation, documentation, training, and other strategies.
- Security teams are increasingly broadening their skills to look at security problems across multiple technologies and systems. They address the full attacker lifecycle, rather than focusing on narrow technical areas (network security, endpoint security, application security, and cloud security, for example). The fact that cloud platforms integrate different technologies closely together amplifies this skill development need.
- The increased rate of change from both technology and security cloud services requires that security processes are continuously updated to keep in sync and manage risk effectively.
- Security threats now reliably bypass network-based security controls, so security teams need to adopt a Zero Trust approach that includes identity, application security, endpoint security, cloud security, CI/CD, user education, and other controls.
- The adoption of DevOps/DevSecOps processes necessitates that security roles be more agile to integrate security natively into the resulting accelerated solution

development lifecycle.

Overview of roles and teams

The following sections provide guidance on which teams and roles typically perform key cloud security functions (when these functions are present in the organization). You should map out your existing approach, look for gaps, and assess whether your organization can and should invest to address those gaps.

The roles that perform security tasks include the following roles.

- Cloud service provider
- Infrastructure/platform teams (architecture, engineering, and operations)
- Security architecture, engineering, and posture management teams:
 - Security architects and engineers (data security, identity and access management (IAM), network security, servers and container security, application security, and DevSecOps)
 - Software security engineers (application security)
 - Posture management (vulnerability management / attack surface management)
- Security operations (SecOps/SOC):
 - Triage analysts (tier 1)
 - Investigation analysts (tier 2)
 - Threat hunting
 - Threat intelligence
 - Detection engineering
- Security Governance, Risk, and Compliance (GRG)
- Security training and awareness

It's critical to ensure that everyone understands their role in security and how to work with other teams. You can accomplish this goal by documenting cross-team security processes and a shared responsibility model for your technical teams. Doing so helps you avoid risk and waste from coverage gaps and from overlapping efforts. It also helps

you avoid common mistakes (antipatterns), like teams selecting weak authentication and cryptography solutions or even attempting to create their own.

Note

A shared responsibility model is similar to a Responsible, Accountable, Consulted, Informed (RACI) model. The shared responsibility model helps illustrate a collaborative approach on who makes decisions and what teams must do to work together for particular items and outcomes.

Cloud service provider

Cloud service providers are effectively virtual team members that provide security functions and capabilities for the underlying cloud platform. Some cloud providers also provide security features and capabilities that your teams can use to manage your security posture and incidents. For more information on what cloud services providers perform, see the [cloud shared responsibility model](#).

Many cloud service providers provide information on their security practices and controls upon request or via a portal like the [Microsoft service trust portal](#) ↗.

Infrastructure/platform teams (architecture, engineering, and operations)

Infrastructure/platform architecture, engineering, and operations teams implement and integrate cloud security, privacy, and compliance controls across the cloud infrastructure and platform environments (across servers, containers, networking, identity, and other technical components).

The engineering and operations roles can focus primarily on cloud or continuous integration and continuous deployment (CI/CD) systems, or they can work across a full range of cloud, CI/CD, on-premises, and other infrastructures and platforms.

These teams are responsible for meeting all the availability, scalability, security, privacy, and other requirements for the organization's cloud services that host business workloads. They work collaboratively with security, risk, compliance, and privacy experts to drive outcomes that blend and balance all these requirements.

Security architecture, engineering, and posture management teams

Security teams work with infrastructure and platform roles (and others) to help translate security strategy, policy, and standards into actionable architectures, solutions, and design patterns. These teams focus on enabling the security success of cloud teams by evaluating and influencing the security of the infrastructure and the processes and tools that are used to manage it. Following are some of the common tasks performed by security teams for the infrastructure:

- **Security architects and engineers** adapt security policies, standards, and guidelines for cloud environments to design and implement controls in partnership with their infrastructure/platform counterparts. Security architects and engineers assist with a broad range of elements, including:
 - *Tenants/subscriptions.* **Security architects and engineers** collaborate with **infrastructure architects and engineers** and **access architects** (identity, networking, app, and others) to help establish security configurations for cloud tenants, subscriptions, and accounts across cloud providers (which are monitored by **security posture management** teams).
 - *IAM.* **Access architects** (identity, networking, app, and others) collaborate with **identity engineers and operations** and infrastructure/platform teams to design, implement, and operate access management solutions. These solutions protect against unauthorized use of the organization's business assets while enabling authorized users to follow business processes to easily and securely access organizational resources. These teams work on solutions like identity directories and single sign-on (SSO) solutions, passwordless and multifactor authentication (MFA), risk-based conditional access solutions, workload identities, privileged identity/access management (PIM/PAM), cloud infrastructure and entitlement management (CIEM), and more. These teams also collaborate with network engineers and operations to design, implement, and operate security service edge (SSE) solutions. Workload teams can take advantage of these capabilities to provide seamless and more secure access to individual workload and application components.
 - *Data security.* **Security architects and engineers** collaborate with **data and AI architects and engineers** to help infrastructure/platform teams establish foundational data security capabilities for all data and advanced capabilities that can be used to classify and protect data in individual workloads. For more information on foundational data security, see the Microsoft security [Data Protection benchmark](#). For more information on protecting data in individual workloads, see the Well-Architected Framework [guidance](#).

- *Network security.* **Security architects and engineers** collaborate with **network architects and engineers** to help infrastructure/platform teams establish foundational network security capabilities like connectivity to the cloud (private/leased lines), remote access strategies and solutions, ingress and egress firewalls, [web application firewalls \(WAFs\)](#), and [network segmentation](#). These teams also collaborate with identity architects, engineers, and operations to design, implement, and operate SSE solutions. Workload teams can take advantage of these capabilities to provide discrete protection or isolation of individual workload and application components.
 - *Servers and container security.* **Security architects and engineers** collaborate with **infrastructure architects and engineers** to help infrastructure/platform teams establish foundational security capabilities for servers, virtual machines (VMs), containers, orchestration/management, CI/CD, and related systems. These teams establish discovery and inventory processes, security baseline/benchmark configurations, maintenance and patching processes, allowlisting for executable binaries, template images, management processes, and more. Workload teams can also take advantage of these foundational infrastructure capabilities to provide security for servers and containers for individual workload and application components.
 - *Software security foundations (for application security and DevSecOps).* **Security architects and engineers** collaborate with **software security engineers** to help infrastructure/platform teams establish application security capabilities that can be used by individual workloads, code scanning, software bill of materials (SBOM) tools, WAFs, and application scanning. See [DevSecOps controls](#) for more information on how to establish a security development lifecycle (SDL). For more information on how workload teams use these capabilities, see the [security development lifecycle](#) guidance in the Well-Architected Framework.
- **Software security engineers** evaluate code, scripts, and other automated logic that's used to manage the infrastructure, including infrastructure as code (IaC), CI/CD workflows, and any other custom-built tools or applications. These engineers should be engaged to protect formal code in compiled applications, scripts, configurations of automation platforms, and any other form of executable code or script that could allow attackers to manipulate the operation of the system. This evaluation might entail simply performing a threat model analysis of a system, or it might involve code review and security scanning tools. See the [SDL practices](#) guidance for more information on how to establish an SDL.
 - **Posture management (vulnerability management / attack surface management)** is the operational security team that focuses on security enablement for technical

operations teams. Posture management helps these teams prioritize and implement controls to block or mitigate attack techniques. Posture management teams work across all technical operations teams (including cloud teams) and often serve as their primary means of understanding security requirements, compliance requirements, and governance processes.

Posture management often serves as a center of excellence (CoE) for security infrastructure teams, similar to the way software engineers often serve as a security CoE for application development teams. Typical tasks for these teams include the following.

- *Monitor security posture.* Monitor all technical systems by using posture management tools like Microsoft Security Exposure Management, Microsoft Entra Permissions Management, non-Microsoft vulnerability and External Attack Surface Management (EASM) and CIEM tools, and custom security posture tools and dashboards. Additionally, posture management performs analysis to provide insights by:
 - *Anticipating highly likely and damaging attack paths.* Attackers "think in graphs" and seek out paths to business-critical systems by chaining together multiple assets and vulnerabilities across different systems (for example, compromise user endpoints, then use the hash/ticket to capture an admin credential, then access the business-critical data). Posture management teams work with security architects and engineers to discover and mitigate these hidden risks, which don't always appear in technical lists and reports.
 - *Conducting security assessments* to review system configurations and operational processes to gain deeper understanding and insights beyond the technical data from security posture tools. These assessments can take the form of informal discovery conversations or formal threat modeling exercises.
- *Assist with prioritization.* Help technical teams proactively monitor their assets and prioritize security work. Posture management helps put the risk mitigation work into context by considering security risk impact (informed by experience, security operations incident reports and other threat intelligence, business intelligence, and other sources) in addition to security compliance requirements.
- *Train, mentor, and champion.* Increase the security knowledge and skills of technical engineering teams through training, mentoring individuals, and informal knowledge transfer. Posture management roles might also work with **organizational readiness / training** and **security education and engagement** roles on formal security training and setting up security within technical teams that evangelize and educate their peers on security.

- *Identify gaps and advocate for fixes.* Identify overall trends, process gaps, tooling gaps, and other insights into risks and mitigations. Posture management roles collaborate and communicate with security architects and engineers to develop solutions, build a case for funding solutions, and assist with rolling out fixes.
- *Coordinate with security operations (SecOps).* Help technical teams work with SecOps roles like detection engineering and threat hunting teams. This continuity across all operational roles helps ensure that detections are in place and implemented correctly, security data is available for incident investigation and threat hunting, processes are in place for collaboration, and more.
- *Provide reports.* Provide timely and accurate reports on security incidents, trends, and performance metrics to senior management and stakeholders to update organizational risk processes.

Posture management teams often evolve from existing software vulnerability management roles to address the full set of functional, configuration, and operational vulnerability types described in the Open Group Zero Trust Reference Model. Each type of vulnerability can allow unauthorized users (including attackers) to take control of software or systems, enabling them to cause damage to business assets.

- *Functional vulnerabilities* occur in software design or implementation. They can allow unauthorized control of the affected software. These vulnerabilities might be flaws in software that your own teams developed or flaws in commercial or open source software (typically tracked by a Common Vulnerabilities and Exposures identifier).
- *Configuration vulnerabilities* are misconfigurations of systems that allow unauthorized access to system functionality. These vulnerabilities can be introduced during ongoing operations, also known as configuration drift. They can also be introduced during the initial deployment and configuration of software and systems, or by weak security defaults from a vendor. Some common examples include:
 - Orphaned objects that allow unauthorized access to items like DNS records and group membership.
 - Excessive administrative roles or permissions to resources.
 - Use of a weaker authentication protocol or cryptographic algorithm that has known security issues.
 - Weak default configurations or default passwords.

- *Operational vulnerabilities* are weaknesses in standard operating processes and practices that allow unauthorized access or control of systems. Examples include:
 - Administrators using shared accounts instead of their own individual accounts to perform privileged tasks.
 - Use of "[browse-up configurations](#)" that create elevation-of-privilege paths that can be abused by attackers. This vulnerability occurs when high-privileged administrative accounts sign in to lower-trust user devices and workstations (like standard user workstations and user-owned devices), sometimes via jump servers that don't effectively mitigate these risks. For more information, see [securing privileged access](#) and [privileged access devices](#).

Security operations (SecOps/SOC)

The SecOps team is sometimes referred to as a Security Operations Center (SOC). The SecOps team focuses on rapidly finding and removing adversary access to the organization's assets. They work in close partnership with technology operations and engineering teams. SecOps roles can work across all technologies in the organization, including traditional IT, operational technology (OT), and Internet of Things (IoT).

Following are the SecOps roles that most often interact with cloud teams:

- **Triage analysts (tier 1).** Responds to incident detections for well-known attack techniques and follows documented procedures to rapidly resolve them (or escalate them to investigation analysts as appropriate). Depending on the SecOps scope and maturity level, this might include detections and alerts from email, endpoint antimalware solutions, cloud services, network detections, or other technical systems.
- **Investigation analysts (tier 2).** Responds to higher-complexity and higher-severity incident investigations that require more experience and expertise (beyond well-documented resolution procedures). This team typically investigates attacks that are conducted by live human adversaries and attacks that affect multiple systems. It works in close partnership with technology operations and engineering teams to investigate incidents and resolve them.
- **Threat hunting.** Proactively searches for hidden threats within the technical estate that have evaded standard detection mechanisms. This role uses advanced analytics and hypothesis-driven investigations.

- **Threat intelligence.** Gathers and disseminates information about attackers and threats to all stakeholders, including business, technology, and security. Threat intelligence teams perform research, share their findings (formally or informally), and disseminate them to various stakeholders, including the cloud security team. This security context helps these teams make cloud services more resilient to attacks because they're using real-world attack information in design, implementation, testing, and operation, and continuously improving.
- **Detection engineering.** Creates custom attack detections and customizes attack detections provided by vendors and the broader community. These custom attack detections supplement vendor-provided detections for common attacks that are commonly found in extended detection and response (XDR) tools and some security information and event management (SIEM) tools. Detection engineers work with cloud security teams to identify opportunities for designing and implementing detections, the data required to support them, and the response/recovery procedures for the detections.

Security Governance, Risk, and Compliance

Security Governance, Risk, and Compliance (GRC) is a set of interrelated disciplines that integrate the technical work of security teams with organizational goals and expectations. These roles and teams can be a hybrid of two or more disciplines or can be discrete roles. Cloud teams interact with each of these disciplines over the course of the cloud technology lifecycle:

- The **governance** discipline is a foundational capability that focuses on ensuring the organization is consistently implementing all aspects of security. Governance teams focus on decision rights (who makes what decisions) and process frameworks that connect and guide teams. Without effective governance, an organization with all the right controls, policies, and technology can still be breached by attackers who found areas where the intended defenses aren't implemented well, fully, or at all.
- The **risk management** discipline focuses on ensuring that the organization is effectively assessing, understanding, and mitigating risk. Risk management roles work with many teams across the organization to create a clear representation of the organization's risk and keep it current. Because many critical business services can be hosted on cloud infrastructure and platforms, cloud and risk teams need to collaborate to assess and manage this organizational risk. Additionally, supply chain security focuses on risks associated with external vendors, open source components, and partners.

- The **compliance** discipline ensures that systems and processes are compliant with regulatory requirements and internal policies. Without this discipline, the organization might be exposed to risk related to noncompliance with external obligations (fines, liability, loss of revenue from inability to operate in some markets, and more). Compliance requirements typically can't keep up with the speed of attacker evolution, but they're an important requirement source nonetheless.

All three of these disciplines operate across all technologies and systems to drive organizational outcomes across all teams. All three also rely on context they get from each other and benefit significantly from current high-fidelity data on threats, business, and the technology environment. These disciplines also rely on architecture to express an actionable vision that can be implemented and security education and policy to establish rules and guide teams through the many daily decisions.

Cloud engineering and operation teams might work with **posture management** roles, **compliance and audit** teams, **security architecture and engineering**, or **chief information security officer (CISO)** roles on GRC topics.

Security education and policy

Organizations must ensure that all roles have basic security literacy and guidance on what they're expected to do regarding security and how to do it. To achieve this goal, you need a combination of written policy and education. The education for cloud teams can be informal mentoring by security professionals who work directly with them, or it can be a formal program with documented curriculum and designated security champions.

In a larger organization, security teams work with **organizational readiness / training** and **security education and engagement** roles on formal security training and setting up security champions within technical teams to evangelize and educate their peers on security.

Security education and policy must help each role understand:

- **Why.** Show each role why security is important to them and their goals in the context of their role responsibilities. If people don't clearly understand why security matters to them, they'll judge it to be unimportant and move on to something else.
- **What.** Summarize what security tasks they need to do in language they already understand. If people don't know what they're being asked to do, they'll assume security isn't important or relevant to them and move on to something else.

- **How.** Ensure that each role has clear instructions on how to apply security guidance in their role. If people don't know how to actually do what they're being asked to do (for example, patch servers, identify whether a link is a phishing link, report a message properly, review code, or perform a threat model), they'll fail and move on to something else.

Example scenario: Typical interoperability among teams

When an organization deploys and operationalizes a WAF, several security teams must collaborate to ensure its effective deployment, management, and integration into the existing security infrastructure. Here's how the interoperability among teams might look in an enterprise security organization:

1. Planning and design

- a. The *governance team* identifies the need for enhanced web application security and allocates budget for a WAF.
- b. The *network security architect* designs the WAF deployment strategy, ensuring it integrates seamlessly with existing security controls and aligns with the organization's security architecture.

2. Implementation

- a. The *network security engineer* deploys the WAF according to the architect's design, configuring it to protect the specific web applications, and enables monitoring.
- b. The *IAM engineer* sets up access controls, ensuring that only authorized personnel can manage the WAF.

3. Monitoring and management

- a. The *posture management team* provides instructions for the SOC to configure monitoring and alerting for the WAF and to set up dashboards to track WAF activity.
- b. The *threat intelligence and detection engineering teams* help to develop response plans for incidents that involve the WAF and to conduct simulations to test these plans.

4. Compliance and risk management

- a. The *compliance and risk management officer* reviews the WAF deployment to ensure it meets regulatory requirements and conducts periodic audits.
- b. The *data security engineer* ensures that the WAF's logging and data protection measures comply with data privacy regulations.

5. Continuous improvement and training

- a. The *DevSecOps engineer* integrates WAF management into the CI/CD pipeline, ensuring that updates and configurations are automated and consistent.
- b. The *security education and engagement specialist* develops and delivers training programs to ensure that all relevant personnel understand how to use and manage the WAF effectively.
- c. The *cloud governance team member* reviews the WAF deployment and management processes to ensure that they align with organizational policies and standards.

By collaborating effectively, these roles ensure that the WAF is deployed correctly and also continuously monitored, managed, and improved to protect the organization's web applications from evolving threats.

Next step

[Integrate security into your cloud adoption strategy](#)

Feedback

Was this page helpful?

 Yes

 No

Integrate security into your cloud adoption strategy

Article • 11/18/2024

Moving your organization to the cloud adds significant complexity to security. To be successful in the cloud, your security strategy needs to meet modern challenges that are inherent to cloud computing. In the adoption and operation of a cloud estate, security becomes a necessary consideration in all facets of the organization. It's not a separate function that's secondarily applied to certain facets, as can be common for organizations that run on-premises technology platforms. When you define your cloud adoption strategy, consider the recommendations provided in this article to ensure that security is an integral part of the strategy and will be built into your cloud adoption plan as you move forward.



This article is a supporting guide to the [Strategy](#) methodology. It describes areas of security optimization that you should consider as you move through that phase in your journey.

Security posture modernization

The strategy of security posture modernization doesn't just involve the adoption of new technologies and new operational practices. It typically also involves a mindset shift across the organization. New [teams and roles](#) might need to be filled, and existing teams and roles might need to be involved in security in ways that they're unaccustomed to. These changes, which can sometimes be monumental for organizations, can be the source of stress and internal conflicts, so it's important to promote healthy, honest, and blame-free communications across the organization throughout the adoption process.

See the [Define a security strategy](#) guide for a comprehensive overview of these considerations.

Adopting Zero Trust as a strategy

Adopting [Zero Trust](#) as a strategy helps you start your cloud journey with a modern approach to security in place. The Zero Trust approach is founded on three principles:

- **Verify explicitly.** Always authenticate and authorize based on all available data points.
- **Use least privilege.** Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.
- **Assume breach.** Minimize the blast radius and segment access. Verify end-to-end encryption, and use analytics to get visibility into activities related to your systems, drive threat detection, and improve defenses.

If you apply these principles across the cloud adoption process, the transformation to modern security can be a smoother experience for the entire organization.

Microsoft provides a [Zero Trust-based security modernization blueprint](#) that organizations can use as a guide. Refer to the [Define strategy phase](#) for strategy recommendations.

Defining a strategy for incident preparedness and response

Establish a clear vision and well-defined, specific objectives for cloud security readiness. Focus on creating security capacity and developing security skills. Align your incident preparedness and response strategy to the overall business strategy to ensure that the business strategy isn't impeded by security. Understand business requirements for reliability and performance to ensure that your strategy can accommodate those requirements while creating the necessary technology foundation to prepare for and respond to incidents.

Defining a strategy for confidentiality

When you define a strategy for adopting confidentiality in an enterprise cloud environment, you need to consider several key points:

- **Prioritize data privacy and protection.** Establish clear business objectives that emphasize the importance of data privacy and protection. These objectives include compliance with relevant regulations like GDPR, HIPAA, and industry standards.
- **Plan for a risk management strategy.** Identify and assess potential risks to data confidentiality and develop strategies to mitigate these risks.

- **Develop a data loss protection (DLP) strategy.** DLP is a set of tools and processes that helps ensure that sensitive data isn't lost, misused, or accessed by unauthorized users. In terms of the principle of confidentiality, it involves defining clear data protection objectives and establishing a framework for implementing robust encryption and access controls. During the strategy phase, DLP is integrated into the overall security vision to help ensure that sensitive data is protected from unauthorized access.

Defining a strategy for integrity

Maintaining data and system integrity requires many of the same strategies as those suggested for confidentiality, like well-designed data protection controls and risk management. These strategies should be augmented with additional considerations for data and system integrity:

- **Prioritize data and system integrity.** Maintaining data and system integrity should be a key tenet in business requirements and objectives. To that end, prioritize security controls and operational practices that support a high level of integrity. In particular, use automation through tooling for as much of your data and system integrity management as is practical. Automation can be used for many functions that are related to integrity, like:
 - *Policy management.*
 - *Data classification and management.*
 - *Infrastructure deployments and update management.*

Defining a strategy for availability

Including considerations for availability in your cloud adoption strategy helps ensure that you're prepared to implement a reliable and resilient cloud estate and that you can be confident about meeting your business requirements as they relate to availability.

Availability requirements and objectives span across the entire cloud estate, including all business functions and workloads and the underlying cloud platform. Ensure that, as you develop your cloud adoption strategy, you start with high-level goals for determining the criticality of various aspects of your cloud estate and begin discussions among stakeholders about what the proper level of availability must be, while still balancing cost and performance requirements and objectives. This approach helps structure your cloud adoption plans so that you can work toward more defined targets as you progress

to the next phases in your cloud adoption journey, laying the groundwork for appropriately scoped designs and standards.

Defining a strategy for sustaining security posture

The journey toward a modern, robust security posture doesn't end with the initial implementation. To keep up with new threats, you need to continuously review and refine your security practices while maintaining strict adherence to standards. Sustaining security is an ongoing effort of running day-to-day operations that meet the expectations your organization has set for itself while preparing for emerging threats and technological changes. Adoption of this principle codifies your continuous improvement approach. It provides security teams with guiding standards for maintaining vigilant security practices and gives stakeholders confidence that security remains a cornerstone tenet of the cloud adoption journey.

When you develop a sustainment strategy, you focus on learning how your overall security strategy performs in the real world and on applying lessons to evolve it continuously. A sustainment strategy should incorporate long-term business goals to ensure that long-term security goals are aligned. When these goals are taken into account, the sustainment strategy defines how the security posture must evolve to stay in alignment.

Example strategy

Your organization should develop your cloud adoption strategy in the way that works best for the organization. The following example shows how you might incorporate the guidance offered in this article into a narrative artifact, like a Word document.

Motivations

The motivation for moving to the cloud is to modernize our line-of-business (LOB) workload and take advantage of the Microsoft worldwide cloud infrastructure to efficiently scale out across the globe as our customer base grows.

Business considerations:

- *Board and senior leadership buy-in.* We must present an executive summary of our cloud adoption plan, with financial projections, to the board for approval. The

executive summary must be co-developed by senior leadership to ensure that the leadership team is in agreement about the high-level plan.

Security considerations:

- *Technical readiness.* Our IT and Security teams will need upskilling to successfully define our migration plan. We might need to add new [teams and roles](#) as we prepare to move to the cloud.

Business outcome: Global reach

We currently operate only in North America. Our five-year plan is to expand into Europe and Asia. Taking advantage of the Microsoft global Azure cloud will allow us to build out the necessary infrastructure to efficiently deliver our LOB application in Europe and Asia.

- **Business owner:** COO
- **Technical owner:** CTO
- **Security owner:** CISO

Business considerations:

- *Budget forecasting.* As part of developing our cloud migration plan, IT, Security, and Sales must co-develop budget forecasting models with the Finance department to ensure that stakeholders understand the potential costs of expanding into Europe and Asia.

Security considerations:

- *Increased attack surfaces.* Expanding across the globe will dramatically increase our attack surfaces by placing publicly exposed systems in multiple regions. We need to rapidly [modernize our security posture](#). We'll follow the [Zero Trust guidance](#) to ensure that we follow best practices.
- *Cloud-focused threats.* Our move to the cloud will bring new threats that we haven't been exposed to. These threats aren't limited to malicious attacks on our systems. The cloud provider is also a major target for threats, and incidents that affect the provider can have downstream effects on our systems or business. We need to review our [incident preparedness and response](#) processes and incorporate necessary improvements as part of our plan.

Business outcome: Data innovation

As our global expansion progresses, our data estate will grow exponentially. Handling that data will be unsustainable unless we adopt cloud-scale data and analytics technologies.

- **Business owner:** CEO
- **Technical owner:** CTO
- **Security owner:** CISO

Business considerations:

- *Local compliance requirements.* We must work with experts on local compliance regulations to ensure that the business is ready to support the technical teams with maintaining compliance. This might mean setting up business entities in certain geographies or using sovereign clouds in countries like Germany and China.

Security considerations:

- *Data confidentiality and integrity at scale.* We must review and improve our **confidentiality** and **integrity** strategies and mechanisms to ensure that, as we adopt new technologies and move into new geographies, we don't put our data or our customers' data at risk of corruption, breach, or loss, and that we comply with regulatory frameworks by default.
- *Zero Trust access and authorization strategy.* We must adopt the Zero Trust approach to ensure that our access and authorization strategy meets modern best practices and is manageable as we expand globally.

Business outcome: Performance and reliability

As we expand across the globe, our LOB workload must maintain the high performance and zero-downtime availability that our customers rely on.

- **Business owner:** COO
- **Technical owner:** CTO
- **Security owner:** CISO

Business considerations:

- *Maintaining performance and reliability throughout the migration.* Our customers have high expectations for our LOB application. We can't afford to suffer

reputational and financial damage if the application experiences downtime or prolonged degraded service over the course of the migration to the cloud. Engaging our Microsoft support team to help design the migration plan and be involved in the migration will minimize the risks of downtime or degraded service.

Security considerations:

- We must develop secure design patterns to ensure that we can efficiently and securely deploy identical infrastructure packages in each new region we expand into. Our [availability](#) strategy should consider tradeoffs that we'll need to make to ensure that security isn't compromised by our performance designs and that our performance targets aren't affected by our security measures.
 - We must include system [integrity](#) processes and mechanisms in our design patterns to ensure that our systems are protected by default when we deploy our workload in new geographies.

Next step

[Plan for a secure cloud adoption](#)

Feedback

Was this page helpful?

 Yes

 No

Plan for a secure cloud adoption

Article • 11/18/2024

Developing a cloud adoption plan can be difficult and often have many technical challenges. You must carefully plan each step of your cloud adoption process, specifically when you update legacy workloads for cloud infrastructure. To build a secure cloud estate from the ground up, you need to integrate security considerations into every phase of your adoption plan. This approach helps ensure that your new cloud environment is secure from the start.

When you make decisions about your migration or implementation, design for the highest security strategy stance that's feasible for your business. Prioritize security over performance and cost efficiency when you start your designs. This approach ensures that you don't introduce risks that could require you to redesign workloads later. The guidance provided in this article can help you develop a cloud adoption plan that has security as a fundamental principle.



This article is a supporting guide to the [Plan](#) methodology. It provides areas of security optimization for you to consider as you move through that phase in your journey.

Plan for landing zone adoption

To build out your cloud estate foundational elements, use the [landing zone](#) approach. This recommendation applies specifically to enterprise and large organizations. Smaller organizations and start-ups might not benefit from adopting this approach at the beginning of their cloud journey. However, it's important to understand the [design areas](#) because you need to include these areas in your cloud adoption plan, even if you don't create a full landing zone.

You can use the Azure landing zone approach to establish a solid foundation for your cloud estate. This foundation helps ensure a manageable environment that you can secure more efficiently according to best practices.

- *Landing Zones:* A landing zone is a preconfigured, enhanced-security, scalable environment in the cloud that serves as a foundation for your workloads. It

includes network topology, identity management, security, and governance components.

- *Benefits:* Landing zones can help you standardize cloud environments. This approach helps ensure consistency and compliance with security policies. Also, they facilitate easier management and scalability.

Security posture modernization

When you develop a security modernization plan, it's essential to focus on adopting new technologies and operational practices. It's equally important that you align these security measures with your business objectives.

Plan for Zero Trust adoption

As you develop your adoption plan, incorporate the principles of Zero Trust across your plan to help structure the phases and steps that teams throughout the organization are responsible for and how they can accomplish their activities.

The Microsoft Zero Trust approach provides guidance for seven technology pillars, including deployment and configuration recommendations. As you build your plan, explore each pillar to help ensure comprehensive coverage of these areas.

Zero Trust technology pillars

- **Identity:** Guidance for verifying identities with strong authentication and controlling access under the principle of least privilege.
- **Endpoints:** Guidance for securing all endpoints, including devices and apps, that interact with your data. This guidance applies regardless of where the endpoints connect from and how they connect.
- **Data:** Guidance for securing all data by using a defense-in-depth approach.
- **Apps:** Guidance for securing the cloud apps and services that you consume.
- **Infrastructure:** Guidance for securing cloud infrastructure through strict policies and enforcement strategies.
- **Network:** Guidance for securing your cloud network through segmentation, traffic inspection, and end-to-end encryption.

- **Visibility, automation, and orchestration:** Guidance for operational policies and practices that help enforce Zero Trust principles.

Business alignment

Alignment between technology and business stakeholders is critical to the success of your security modernization plan. You must approach plan development as a collaborative process and negotiate with stakeholders to find the best way to adapt processes and policies. Business stakeholders must understand how the modernization plan affects business functions. Technology stakeholders must know where to make concessions to keep critical business functions secure and intact.

Incident preparedness and response

- **Plan for preparedness:** Plan for incident preparation by evaluating vulnerability management solutions, threat and incident detection systems, and robust infrastructure monitoring solutions. Plan for infrastructure hardening to reduce attack surfaces.
- **Plan for incident response:** Build a robust incident response plan to help ensure cloud security. In the Plan phase, start drafting your incident response plan by identifying the roles and key phases, such as investigation, mitigation, and communications. You'll add details about these roles and phases as you create your cloud estate.

Plan for confidentiality

- **Data loss protection:** To establish organizational data confidentiality across the enterprise, meticulously plan specific data loss prevention policies and procedures. This process includes identifying sensitive data, determining how to protect the data, and planning for the deployment of encryption technologies and secure access controls.
- **Include data protection requirements in your cloud migration or development plans:**
 - *Data classification:* Identify and classify data based on sensitivity and regulatory requirements. This process helps you apply appropriate security measures.
 - *Encryption:* Ensure that data is encrypted at rest and in transit. Use strong encryption standards to protect sensitive information.

- **Access controls:** Implement strict access controls to help ensure that only authorized users can access sensitive data. Use multifactor authentication and role-based access control. Follow the principle of Zero Trust and verify explicitly, always authenticating and authorizing based on all available data points. These data points include user identity, location, device health, service or workload, data classification, and anomalies.

Plan for integrity

In addition to the measures recommended for confidentiality, consider implementing specific data and system integrity measures.

- **Plan for data and system integrity observability and governance:** In your cloud adoption or development plans, include plans to monitor data and systems for unauthorized changes and policies for data hygiene.
- **Plan for integrity incidents:** In your incident response plan, include considerations for integrity. These considerations should address unauthorized changes to data or systems and how to remediate invalid or corrupted data discovered through your monitoring and data hygiene practices.

Plan for availability

Your cloud adoption plan should address availability by adopting standards for architecture design and operations. These standards guide the implementation and future phases and provide a blueprint for how you can achieve availability requirements. Consider the following recommendations as you build out your cloud adoption plan:

- **Standardize infrastructure and application design patterns:** Standardize infrastructure and application design patterns to help ensure that your workloads are reliable. Avoid unnecessary complexity to make designs repeatable and to discourage shadow IT behaviors. Follow best practices for [highly available infrastructure](#) and [resilient applications](#) as you define your design standards.
- **Standardize development tools and practices:** Develop well-defined and enforceable [standards for your development tools and practices](#). This approach helps ensure that your deployments adhere to the principles of the CIA Triad and incorporates best practices for [safe deployments](#).
- **Standardize operational tools and practices:** Depend on well-defined and strictly enforced standards for operators to follow in order to maintain confidentiality, integrity, and availability. Follow standards consistently and train on them routinely

so that your systems are resilient to attacks and can respond efficiently to incidents.

Plan for security sustainment

For the long-term sustainment of your security posture, adopt a mindset of continuous improvement across the organization. This approach includes not only adhering to operational standards in everyday practices but also actively seeking opportunities for enhancement. Regularly review your standards and policies, and implement a training program that fosters a continuous improvement mindset.

In order to plan your security baseline, you must first understand your current security posture to establish your baseline. Use an automated tool like [Microsoft Secure Score](#) to establish your baseline quickly and gain insights into areas for improvement.

Next step

[Prepare your secure cloud estate](#)

Feedback

Was this page helpful?

 Yes

 No

Prepare your secure cloud estate

Article • 11/18/2024

During the Ready phase of a cloud adoption journey, you focus on creating the foundation of the estate. The Microsoft [Azure landing zone](#) approach provides enterprises and large organizations with a more secure, scalable, modular design pattern to follow when they implement their estates. Smaller organizations and startups might not need the level of organization that the landing zone approach provides, but an understanding of the landing zone philosophy can help any organization strategize a foundational design and gain a high degree of security and scalability.

After you define your cloud adoption [strategy](#) and [plan](#), you can begin the implementation phase by designing the foundation. Use the recommendations in this guide to ensure that your foundation design and implementation prioritize security.



This article is a supporting guide to the [Ready](#) methodology. It describes areas of security optimization that you should consider as you move through that phase in your journey.

Security posture modernization

The first implementation steps in modernizing your security posture are building your landing zone or cloud foundation and creating or modernizing your identity, authorization, and access platform.

- **Adopting the landing zone approach:** Adopting the landing zone approach or incorporating the design principles of the landing zone approach to the extent practical for your use case allows you to start your implementation in an optimized way. As your cloud estate evolves, keeping different domains of your estate separated helps keep the entire estate more secure and manageable.
 - If you don't plan to adopt a complete enterprise landing zone, you still need to understand the [design areas](#) and apply guidance that's relevant to your cloud estate. You need to think about all of these design areas and implement controls that are specific to each area, no matter how your foundation is

architected. For example, using management groups can help you govern your cloud estate even if it only consists of a few subscriptions.

Develop secure, scalable landing zones that provide controlled environments for deploying cloud resources. These zones help you ensure that security policies are consistently applied and that resources are segregated according to their security requirements. See the [security design area](#) for detailed guidance on this topic.

- **Modern identity, authorization, and access:** Based on the principles of Zero Trust, the modern approach to identity, authorization, and access moves from trust-by-default to trust-by-exception. It follows from these principles that users, devices, systems, and apps should be allowed to access only resources that they require, and only for as long as needed to meet their needs. The same guidance applies to the foundational elements of your estate: tightly control permissions to subscriptions, networking resources, governance solutions, the identity and access management (IAM) platform, and tenants by following the same recommendations that you follow for the workloads that you run. See the [identity and access management design area](#) for detailed guidance on this topic.

Azure facilitation

- **Azure landing zone accelerators:** Microsoft maintains several landing zone accelerators, which are prepackaged deployments of a given workload type that can be easily deployed into a landing zone to quickly get you started. They include accelerators for [Azure Integration Services](#), [Azure Kubernetes Service \(AKS\)](#), [Azure API Management](#), and others. See the [Modern application platform scenario](#) section of the Cloud Adoption Framework for Azure documentation for a full list of accelerators and other topics related to modern application considerations.
- **Azure landing zones Terraform module:** You can optimize your landing zone deployments with automation by using the [Azure landing zones Terraform module](#). By using your continuous integration and continuous deployment (CI/CD) pipeline to deploy landing zones, you can ensure that all of your landing zones are deployed identically, with all security mechanisms in place.
- **Microsoft Entra:** [Microsoft Entra](#) is a family of identity and network access products. It enables organizations to implement a Zero Trust security strategy and create a [trust fabric](#) that verifies identities, validates access conditions, checks permissions, encrypts connection channels, and monitors for compromise.

Prepare for incident preparedness and response

After you define your strategy and develop your plan for incident preparedness and response, you can begin your implementation. Whether you adopt a full enterprise landing zone design or a smaller foundational design, network segregation is critical for maintaining a high degree of security.

Network segmentation: Design a network architecture with proper segmentation and isolation to minimize attack surfaces and contain potential breaches. Use techniques like virtual private clouds (VPCs), subnets, and security groups to manage and control traffic. See the [Plan for network segmentation](#) article for detailed guidance on this topic. Be sure to review the rest of the Azure landing zone network security guides. The guidance includes recommendations for [inbound and outbound connectivity](#), [network encryption](#), and [traffic inspection](#).

Azure facilitation

- **Azure Virtual WAN:** [Azure Virtual WAN](#) is a networking service that consolidates many networking, security, and routing functionalities to provide a single operational interface. The design is a hub-and-spoke architecture that has scale and performance built in for branches (VPN/SD-WAN devices), users (Azure VPN/OpenVPN/IKEv2 clients), Azure ExpressRoute circuits, and virtual networks. When you implement your landing zones, Azure Virtual WAN can help you optimize your network through segmentation and security mechanisms.

Prepare for confidentiality

During the Ready phase, preparing for your workloads from a confidentiality standpoint is a process of ensuring that your IAM policies and standards are implemented and enforced. This preparation ensures that, when you deploy workloads, your data will be secured by default. Be sure to have well-governed policies and standards for:

- *The principle of least privilege.* Grant users the minimum access required to perform their tasks.
- *Role-based access control (RBAC).* Assign roles and permissions based on job responsibilities. Doing so helps you manage access efficiently and reduces the risk of unauthorized access.
- *Multifactor authentication (MFA).* Implement MFA to add an extra layer of security.

- *Conditional access controls.* Conditional access controls provide additional security by enforcing policies based on specific conditions. Policies can include enforcing MFA, blocking access based on geography, and many other scenarios. When you choose an IAM platform, be sure that conditional access is supported and that the implementation meets your requirements.

Azure facilitation

- [Microsoft Entra Conditional Access](#) is the Microsoft Zero Trust policy engine. It takes signals from various sources into account when enforcing policy decisions.

Prepare for integrity

As with your confidentiality preparations, ensure that you have well-governed policies and standards for data and system integrity so you deploy workloads with improved security by default. Define policies and standards for the following areas.

Data management practices

- *Data classification:* Create a data classification framework and sensitivity-label taxonomy that defines high-level categories of data security risk. You'll use that taxonomy to simplify everything from data inventory or activity insights, to policy management, to investigation prioritization. See [Create a well-designed data classification framework](#) for detailed guidance on this topic.
- *Data verification and validation:* Invest in tooling that automates data verification and validation to reduce the burden on your data engineers and administrators and to decrease the risk of human error.
- *Backup policies:* Codify backup policies to ensure that all data is regularly backed up. Test backups and restores regularly to ensure that backups succeed and that data is correct and consistent. Align these policies with your organization's recovery time objective (RTO) and recovery point objective (RPO) targets.
- *Strong encryption:* Ensure that your cloud provider encrypts your data at rest and in transit by default. On Azure, your data is encrypted end to end. See the [Microsoft Trust Center](#) for details. For the services that you use in your workloads, ensure that strong encryption is supported and appropriately configured to meet your business requirements.

System integrity design patterns

- *Security monitoring*: To detect unauthorized changes to your cloud systems, design a robust security monitoring platform as part of your overall monitoring and observability strategy. See the Manage methodology [monitoring section](#) for detailed overall guidance. See the Zero Trust [Visibility, automation, and orchestration](#) guide for recommendations on security monitoring.
 - *SIEM and threat detection*: Use security information and event management (SIEM) and security orchestration, automation, and response (SOAR) tooling and threat detection tooling to detect suspicious activities and potential threats to your infrastructure.
- *Automated configuration management*: Codify the use of tooling to automate configuration management. Automation helps you ensure that all system configurations are consistent, free from human error, and enforced automatically.
- *Automated patch management*: Codify the use of tooling for managing and governing updates for virtual machines. Automated patching helps ensure that all systems are patched regularly and that system versions are consistent.
- *Automated infrastructure deployments*: Codify the use of infrastructure as code (IaC) for all deployments. Deploy IaC as part of your CI/CD pipelines. Apply the same [safe deployment practices](#) for IaC deployments as you would for software deployments.

Azure facilitation

- [Azure Policy](#) and [Microsoft Defender for Cloud](#) work together to help you define and enforce security policies across your cloud estate. Both solutions support the governance of your foundational elements and your workload resources.
- [Azure Update Manager](#) is the native Azure update and patch management solution. You can extend it to on-premises systems and Arc-enabled systems.
- [Microsoft Sentinel](#) is the Microsoft SIEM and SOAR solution. It provides cyberthreat detection, investigation, and response, proactive hunting, and a comprehensive view across your enterprise.

Prepare for availability

Designing your workloads for resiliency helps ensure that your business can withstand malfunctions and security incidents, and that operations can continue while problems with affected systems are addressed. The following recommendations, which align to Cloud Adoption Framework principles, can help you design resilient workloads:

- **Implement resilient application design.** Adopt application design patterns that enhance resilience against both infrastructure and non-infrastructure incidents, aligning with the broader principles of the Cloud Adoption Framework. Standardize on designs that incorporate self-healing and self-preservation mechanisms to ensure continuous operation and rapid recovery. For detailed guidance on resilient design patterns, see the Well-Architected Framework's [Reliability](#) pillar.
- **Adopt serverless architecture.** Use serverless technologies, including platform as a service (PaaS), software as a service (SaaS), and function as a service (FaaS), to reduce server management overhead, automatically scale with demand, and improve availability. This approach supports the Cloud Adoption Framework emphasis on modernizing workloads and optimizing operational efficiency.
- **Use microservices and containerization.** Implement microservices and containerization to avoid monolithic applications by breaking them down into smaller, independent services that you can deploy and scale independently. This approach aligns with Cloud Adoption Framework principles of agility and scalability in cloud environments.
- **Decouple services.** Strategically isolate services from each other to reduce the blast radius of incidents. This strategy helps to ensure that failures in one component don't affect the entire system. It supports the Cloud Adoption Framework governance model by promoting robust service boundaries and operational resilience.
- **Enable automatic scaling.** Ensure that your application architecture supports automatic scaling to handle varying loads so that it can maintain availability during traffic spikes. This practice aligns with Cloud Adoption Framework guidance on creating scalable and responsive cloud environments and can help you keep costs manageable and predictable.
- **Implement fault isolation.** Design your application to isolate failures to individual tasks or functions. Doing so can help prevent widespread outages and enhance resilience. This approach supports the Cloud Adoption Framework focus on creating reliable and fault-tolerant systems.
- **Ensure high availability.** Incorporate built-in redundancy and disaster recovery mechanisms to maintain continuous operation. This approach supports Cloud Adoption Framework best practices for high availability and business continuity planning.

- **Plan for automatic failover.** Deploy applications across multiple regions to support seamless failover and uninterrupted service. This approach aligns with the Cloud Adoption Framework strategy for geographic redundancy and disaster recovery.

Prepare for security sustainment

During the Ready phase, preparing for long-term security sustainment involves ensuring that the foundational elements of your estate adhere to security best practices for the initial workloads but are also scalable. Doing so helps you ensure that, as your estate grows and evolves, your security won't be compromised and the management of your security won't become too complex and burdensome. This, in turn, helps you avoid shadow IT behaviors. To that end, during the Ready phase, think about how your business goals for the longer term can be accomplished without major architectural redesigns or major overhauls to operational practices. Even if you choose to establish a much simpler foundation than a landing zone design, ensure that you can transition your foundational design to an enterprise architecture without needing to redeploy major elements of your estate, like networking and critical workloads. Creating a design that can grow as your estate grows, but still remain secure, is instrumental to the success of your cloud journey.

See [Transition an existing Azure environment to the Azure landing zone conceptual architecture](#) for recommendations on moving an existing Azure footprint into a landing zone architecture.

Next step

[Perform your cloud adoption with enhanced security](#)

Feedback

Was this page helpful?

 Yes

 No

Perform your cloud adoption securely

Article • 11/18/2024

When you implement your cloud estate and migrate workloads in, it's essential to establish robust security mechanisms and practices. This approach ensures that your workloads are secure from the start and prevents the need to address security gaps after the workloads are in production. Prioritize security during the Adopt phase to ensure that workloads are built consistently and according to best practices. Established security practices also prepare IT teams for cloud operations through well-designed policies and procedures.

Whether you're migrating workloads into the cloud or building an entirely new cloud estate, you can apply the guidance in this article. The Cloud Adoption Framework [Adopt](#) methodology incorporates the [Migrate](#), [Modernize](#), [Innovate](#), and [Relocate](#) scenarios. Regardless of the path that you take during the Adopt phase of your cloud journey, it's important to consider the recommendations in this article as you establish the foundational elements of your cloud estate and build or migrate workloads.



This article is a supporting guide to the [Adopt](#) methodology. It provides areas of security optimization that you should consider as you move through that phase in your journey.

Security posture modernization adoption

Consider the following recommendations as you work toward modernizing your security posture as part of the Adopt phase:

- **Security baselines:** Define security baselines that include availability requirements to establish a clear and robust foundation for development. To save you time and reduce the risk of human error in analyzing your environments, use an off-the-shelf security baseline analysis tool.
- **Embrace automation:** Use automation tools to manage routine tasks to reduce the risk of human error and improve consistency. Take advantage of cloud services that can automate failover and recovery procedures. Tasks that you might consider automating include:

- Infrastructure deployments and management
- Software development lifecycle activities
- Testing
- Monitoring and alerting
- Scaling
- **Zero Trust access and authorization controls:** Implement strong access controls and identity management systems to ensure that only authorized personnel have access to critical systems and data. This approach reduces the risk of malicious activities that could disrupt services. Standardize strictly enforced role-based access controls (RBACs) and require multifactor authentication to prevent unauthorized access that could disrupt service availability. For more information, see [Securing identity with Zero Trust](#).

Change management institutionalization

Effective adoption and change management (ACM) methodologies are crucial for ensuring the successful implementation and institutionalization of access controls. Some of the best practices and methodologies include:

- **Prosci ADKAR Model:** This model focuses on five key building blocks for successful change. These components are Awareness, Desire, Knowledge, Ability, and Reinforcement. By addressing each element, organizations can ensure that employees understand the need for access controls, are motivated to support the change, have the necessary knowledge and skills, and receive ongoing reinforcement to maintain the change.
- **Kotter's 8-Step Change Model:** This model outlines eight steps for leading change. These steps include creating a sense of urgency, forming a powerful coalition, developing a vision and strategy, communicating the vision, empowering employees for broad-based action, generating short-term wins, consolidating gains, and anchoring new approaches into the culture. By following these steps, organizations can effectively manage the adoption of access controls.
- **Lewin's Change Management Model:** This model has three stages, which are Unfreeze, Change, and Refreeze. In the Unfreeze stage, organizations prepare for change by identifying the need for access controls and creating a sense of urgency. In the Change stage, new processes and practices are implemented. In the Refreeze stage, the new practices are solidified and integrated into the organizational culture.

- **Microsoft Adoption and change management framework:** This framework provides a structured approach to driving adoption and change by defining success criteria, engaging stakeholders, and preparing the organization. This framework also measures success to ensure the effectiveness of the implementation. It emphasizes the importance of communication, training, and support to ensure that access controls are effectively adopted and institutionalized.

Organizations can ensure that access controls are implemented and embraced by employees by incorporating these ACM methodologies and best practices. This approach results in a more secure and compliant enterprise environment.

Azure facilitation

- **Establish a security baseline:** [Microsoft Secure Score](#) can help you establish baselines with tangible recommendations for improvements. It's provided as part of the Microsoft Defender XDR suite and can analyze the security of many [Microsoft and non-Microsoft products](#).
- **Infrastructure deployment automation:** [Azure Resource Manager templates \(ARM templates\)](#) and [Bicep](#) are Azure-native tools for deploying infrastructure as code (IaC) by using declarative syntax. ARM templates are written in JSON, whereas Bicep is a domain-specific language. You can easily integrate both into [Azure Pipelines](#) or [GitHub Actions](#) continuous integration and continuous delivery (CI/CD) pipelines.
 - [Terraform](#) is another declarative IaC tool that's fully supported in Azure. You can use Terraform to deploy and manage infrastructure, and you can integrate it into your CI/CD pipeline.
 - You can use Microsoft Defender for Cloud to [discover misconfigurations in IaC](#).
 - **Azure Deployment Environments:** [Deployment Environments](#) enables development teams to quickly create consistent app infrastructure by using project-based templates. These templates minimize setup time and maximize security, compliance, and cost efficiency. A deployment environment is a collection of Azure resources that are deployed in predefined subscriptions. Development infrastructure administrators can enforce enterprise security policies and provide a curated set of predefined IaC templates.

Development infrastructure administrators define deployment environments as catalog items. Catalog items are hosted in a GitHub or Azure DevOps repository, called a *catalog*. A catalog item consists of an IaC template and a manifest.yml file.

You can script the creation of deployment environments and programmatically manage the environments. For detailed, workload-focused guidance, see the Azure Well-Architected Framework's [IaC approach](#).

- **Routine task automation:**

- **Azure Functions:** [Azure Functions](#) is a serverless tool that you can use to automate tasks by using your preferred development language. Functions provides a comprehensive set of event-driven triggers and bindings that connect your functions to other services. You don't have to write extra code.
- **Azure Automation:** PowerShell and Python are popular programming languages for automating operational tasks. Use these languages to perform operations like restarting services, transferring logs between data stores, and scaling infrastructure to meet demand. You can express these operations in code and run them on demand. Individually, these languages lack a platform for centralized management, version control, or tracking run history. The languages also lack a native mechanism for responding to events like monitoring-driven alerts. To provide these capabilities, you need an automation platform. [Automation](#) provides an Azure-hosted platform for hosting and running PowerShell and Python code across cloud and on-premises environments, including Azure and non-Azure systems. PowerShell and Python code is stored in an Automation runbook. Use Automation to:
 - Trigger runbooks on demand, on a schedule, or through a webhook.
 - Run history and logging.
 - Integrate a secrets store.
 - Integrate source control.
- **Azure Update Manager:** [Update Manager](#) is a unified service that you can use to manage and govern updates for virtual machines. You can monitor Windows and Linux update compliance across your workload. You can also use Update Manager to make real-time updates or schedule them within a defined maintenance window. Use Update Manager to:
 - Oversee compliance on your entire fleet of machines.
 - Schedule recurring updates.
 - Deploy critical updates.

- **Azure Logic Apps and Microsoft Power Automate:** When you build custom digital process automation (DPA) to handle workload tasks like approval flows or building ChatOps integrations, consider using [Logic Apps](#) or [Power Automate](#). You can construct workflows from built-in connectors and templates. Logic Apps and Power Automate are built on the same underlying technology and are well-suited for trigger-based or time-based tasks.
- **Automatic scaling:** Many Azure technologies have built-in automatic scaling capabilities. You can also program other services to automatically scale by using APIs. For more information, see [Autoscaling](#).
- **Azure Monitor action groups:** To automatically run self-healing operations when an alert is triggered, use [Azure Monitor action groups](#). You can define these operations by using a runbook, an Azure function, or a webhook.

Incident preparedness and response adoption

After you establish your landing zone or other platform design with secure network segmentation and well-designed subscription and resource organization, you can begin implementation with a focus on incident preparedness and response. During this phase, developing your preparedness and response mechanisms, including your incident response plan, ensures that your cloud estate and operational practices align with business goals. This alignment is crucial for maintaining efficiency and achieving strategic objectives. The adoption phase should approach incident preparedness and response from two perspectives. These perspectives are threat readiness and mitigation, and infrastructure and application security.

Threat readiness and mitigation

- **Threat detection:** Implement advanced monitoring tools and practices to detect threats in real-time. This implementation includes setting up alert systems for unusual activities and integrating extended detection and response (XDR) and security information and event management (SIEM) solutions. For more information, see [Zero Trust threat protection and XDR](#).
- **Vulnerability management:** Identify and mitigate vulnerabilities regularly through patch management and security updates to ensure that systems and applications are protected against known threats.
- **Incident response:** Develop and maintain an incident response plan that includes detection, analysis, and remediation steps to quickly address and recover from security incidents. For workload-focused guidance, see [Recommendations for](#)

[security incident response](#). Automate mitigation activities as much as possible to make these activities more efficient and less prone to human error. For example, if you detect a SQL injection, you can have a runbook or workflow that automatically locks all connections to SQL to contain the incident.

Infrastructure and application security

- **Secure deployment pipelines:** Build CI/CD pipelines with integrated security checks to ensure that applications are securely developed, tested, and deployed. This solution includes static code analysis, vulnerability scanning, and compliance checks. For more information, see [Zero Trust developer guidance](#).
- **IaC deployments:** Deploy all infrastructure through code, without exception. Reduce the risk of misconfigured infrastructure and unauthorized deployments by mandating this standard. Colocate all IaC assets with application code assets and apply the same [safe deployment practices](#) as software deployments.

Azure facilitation

- **Threat detection and response automation:** Automate threat detection and response with the automated investigation and response functionality in [Microsoft Defender XDR](#).
- **IaC deployment security:** Use [deployment stacks](#) to manage Azure resources as a single, cohesive unit. Prevent users from performing unauthorized modifications by using [deny settings](#).

Adopt the principle of confidentiality

After the overarching strategy and implementation plan for adopting the CIA Triad principle of confidentiality is already in place, the next step is to focus on ACM. This step includes ensuring that encryption and secure access controls are effectively implemented and institutionalized across the enterprise cloud environment. In the adoption phase, data loss prevention (DLP) measures are implemented to protect sensitive data in transit and data at rest. The implementation involves deploying encryption solutions, configuring access controls, and training all employees on the importance of data confidentiality and adherence to DLP policies.

Implement encryption and secure access controls

To protect sensitive information from unauthorized access, it's crucial that you implement robust encryption and secure access controls. Encryption ensures that data is unreadable to unauthorized users, while access controls regulate who can access specific data and resources. Understand the encryption capabilities of the cloud services that you deploy and enable the appropriate encryption mechanisms to meet your business requirements.

Incorporate and adopt associated standards

To ensure the consistent implementation of encryption and access controls, it's essential to develop and adopt associated standards. Organizations should establish clear guidelines and best practices for using encryption and access controls, and ensure that these standards are communicated to all employees. For example, a standard might specify that all sensitive data must be encrypted by using AES-256 encryption, and that access to this data must be restricted to authorized personnel only. Organizations can ensure that encryption and access controls are consistently applied across the enterprise by incorporating these standards into their policies and procedures. Providing regular training and support further reinforces these practices among employees. Other examples include:

- **Strong encryption:** Enable encryption on data stores when possible and consider managing your own keys. Your cloud provider might offer encryption at rest for the storage that your data store is hosted on, and give you the option of enabling database encryption like [transparent data encryption](#) in Azure SQL Database. Apply the extra layer of encryption when possible.
- **Access controls:** Apply RBAC, conditional access controls, just-in-time access, and just-enough-access to all data stores. Standardize the practice of reviewing permissions regularly. Restrict write access to configuration systems, which allows changes only through a designated automation account. This account applies modifications after thorough review processes, typically as part of Azure Pipelines.
- **Standards adoption:** The organization might develop a standard that requires all emails that contain sensitive information to be encrypted by using [Microsoft Purview Information Protection](#). This requirement ensures that sensitive data is protected during transmission and only accessible by authorized recipients.

Azure facilitation

- **SIEM and SOAR solutions:** [Microsoft Sentinel](#) is a scalable, cloud-native SIEM that delivers an intelligent and comprehensive solution for SIEM and security

orchestration, automation, and response (SOAR). Microsoft Sentinel provides threat detection, investigation, response, and proactive hunting, with a high-level overview of your enterprise.

- **Azure encryption:** Azure provides encryption for services like Azure SQL Database, Azure Cosmos DB, and Azure Data Lake. The supported encryption models include server-side encryption with service-managed keys, customer-managed keys in Azure Key Vault, and customer-managed keys on customer-controlled hardware. Client-side encryption models support data encryption by an application before it's sent to Azure. For more information, see [Azure encryption overview](#).
- **Access control management:** Formerly known as Azure Active Directory, [Microsoft Entra ID](#) provides comprehensive identity and access management capabilities. It supports multifactor authentication, conditional access policies, and single sign-on to ensure that only authorized users can access sensitive data.
 - [Microsoft Entra ID Protection](#) uses advanced machine learning to identify sign-in risks and unusual user behavior to block, challenge, limit, or grant access. It helps prevent identity compromise, protects against credential theft, and provides insights into your identity security posture.
 - [Microsoft Defender for Identity](#) is a cloud-based security identity threat detection solution that helps secure your identity monitoring across your organization. It can help you better identify, detect, and investigate advanced threats directed at your organization through automated threat detection and response mechanisms.
- **Azure confidential computing:** This service protects data while it's being processed. It uses hardware-based trusted execution environments to isolate and protect data in use, ensuring that even cloud administrators can't access the data.

Adopt the principle of integrity

In the Adopt phase, planning and designs are turned into real-world implementations. To ensure data and system integrity, build your systems according to the standards that you developed in earlier phases. Additionally, train engineers, administrators, and operators on the relevant protocols and procedures.

Data integrity adoption

- **Data classification:** Implement your data classification framework through automation when possible and manually when necessary. Use off-the-shelf tools to

automate your data classification and identify sensitive information. Manually label documents and containers to ensure accurate classification. Curate data sets for analytics by taking advantage of the expertise of knowledgeable users to establish sensitivity.

- **Data verification and validation:** Take advantage of built-in verification and validation functionality in the services that you deploy. For example, Azure Data Factory has built-in functionality to [verify data consistency](#) when you move data from a source to a destination store. Consider adopting practices like:
 - Using the CHECKSUM and BINARY_CHECKSUM functions in SQL to ensure that data isn't corrupted in transit.
 - Storing hashes in tables and creating subroutines that modify the hashes when the last modified date changes.
- **Monitoring and alerting:** Monitor your data stores for changes with detailed change history information to help with reviews. Configure alerting to ensure that you have appropriate visibility and can take efficient actions if there are any incidents that might affect data integrity.
- **Backup policies:** Apply backup policies on all appropriate systems. Understand the backup capabilities of platform as a service and software as a service services. For example, Azure SQL Database includes [automatic backups](#), and you can configure the retention policy as necessary.
- **Share design standards:** Publish and share application design standards that incorporate data integrity mechanisms across the organization. Design standards should encompass nonfunctional requirements, such as natively tracking configuration and data changes at the application level and recording this history within the data schema. This approach mandates that the data schema retains details about data history and configuration history as part of the datastore, in addition to standard logging mechanisms to strengthen your integrity monitoring.

System integrity adoption

- **Security monitoring:** Use a robust monitoring solution to automatically enroll all resources in your cloud estate and ensure that alerting is enabled and configured to notify appropriate teams when incidents occur.
- **Automated configuration management:** Deploy and configure a configuration management system that automatically enrolls new systems and manages your configurations continuously.

- **Automated patch management:** Deploy and configure a patch management system that automatically enrolls new systems and manages patching according to your policies. Prefer native tooling to your cloud platform.

Azure facilitation

- **Data classification and labeling:** [Microsoft Purview](#) is a robust set of solutions that can help your organization govern, protect, and manage data, wherever it lives. It offers manual and automatic [data classification](#) and [sensitivity labeling](#).
- **Configuration management:** [Azure Arc](#) is a centralized and unified infrastructure governance and management platform that you can use to manage configurations for cloud-based and on-premises systems. By using Azure Arc, you can extend your security baselines from [Azure Policy](#), your [Defender for Cloud](#) policies, and Secure Score evaluations, as well as logging and monitoring all your resources in one place.
- **Patch management:** [Azure Update Manager](#) is a unified update management solution for Windows and Linux machines that you can use for Azure, on-premises, and multicloud environments. It has built-in support for [Azure Policy](#) and [Azure Arc](#) managed machines.

Adopt the principle of availability

After the resilient design patterns are defined, your organization can move on to the adoption phase. For detailed guidance on workload availability, refer to the Well-Architected Framework's [Reliability](#) pillar and the [Azure reliability](#) documentation. In the context of cloud adoption, the focus is on establishing and codifying operational practices that support availability.

Establish operational practices to support availability

To maintain a highly available cloud estate, teams that operate the cloud systems must adhere to standardized, mature practices. These practices should include:

- **Operational continuity:** Organizations must plan for continuous operations even under attack conditions. This approach includes establishing processes for rapid recovery and maintaining critical services at a degraded level until full recovery is possible.
- **Robust and continuous observability:** An organization's ability to detect security incidents as they happen allows them to initiate their incident response plans

quickly. This strategy helps minimize the business effects as much as possible.

Incident detection is only possible through a well-designed monitoring and alerting system, which follows best practices for threat detection. For more information, see the [observability guide](#) and [Security monitoring and threat detection guide](#).

- **Proactive maintenance:** Standardize and enforce system updates through policies. Schedule regular maintenance windows to apply updates and patches to systems without disrupting services. Conduct regular health checks and maintenance activities to ensure that all components are functioning optimally.
- **Standardized governance policies:** Enforce all security standards through tooling-supported policies. Use a policy management tool to ensure that all of your systems are compliant with your business requirements by default and that your policies are easily auditable.
- **Disaster recovery preparedness:** Develop and regularly test disaster recovery plans for your workloads to ensure that they're recoverable if a disaster occurs. For more information, see [Disaster recovery](#). Automate recovery activities as much as possible. For example, use automatic failover capabilities in services like [Azure SQL Database](#).
- **Service-level agreements:** Service-level agreements (SLAs) that your cloud platform provides for their services help you understand the guaranteed uptime for the components of your workloads. Use those SLAs as your basis to then develop your own target metrics for the SLAs that you provide to your customers. Microsoft publishes the SLAs for all cloud services at [SLAs for online services](#) .
- **Compliance requirements:** Adhere to regulations such as the General Data Protection Regulation (GDPR) and HIPAA to ensure that systems are designed and maintained to high standards, including standards that are related to availability. Noncompliance can result in legal actions and fines that might disrupt business operations. Compliance often isn't limited to system configuration. Most compliance frameworks also require risk management and incident response standards. Ensure that your operational standards meet the framework requirements and that staff is trained regularly.

Azure facilitation

Policy and compliance management:

- [Azure Policy](#) is a policy management solution that helps enforce organizational standards and assess compliance at-scale. To automate policy enforcement for

many Azure services, take advantage of [built-in policy definitions](#).

- **Defender for Cloud** provides security policies that can automate compliance with your security standards.
- **Operational continuity and disaster recovery:** Many Azure services have built-in recovery capabilities that you can incorporate into your operational continuity and disaster recovery plans. For more information, see [Azure services reliability guides](#).

Adopt security sustainment

Consider the following recommendations to help ensure that the security mechanisms and practices that you put in place as part of your cloud adoption can be sustained and continuously improved as you continue your journey:

- **Institute a security review board:** Create a security review board that continuously reviews projects and mandates security controls. Review your processes regularly to find areas of improvement. Develop processes to ensure that security is always top of mind for everyone.
- **Implement a vulnerability management solution:** Use a vulnerability management solution to monitor the security vulnerability risk score and have a process defined to act on the highest risk score to lowest to minimize the risk. Track the latest common vulnerabilities and exposures risks. Have a policy to apply those mitigations regularly for remediation.
- **Harden production infrastructure:** Secure your cloud estate by hardening your infrastructure. To harden your infrastructure according to industry best practices, follow benchmarking guidance like the [Center for Internet Security \(CIS\) benchmarks](#).
- **Use the MITRE ATT&CK knowledge base:** Use the [MITRE ATT&CK](#) knowledge base to help develop threat models and methodologies for common real-world attack tactics and techniques.
- **Shift left:** Use segregated environments with different access levels for preproduction versus production. This approach helps you shift left, which adds security concerns to all phases of development and provides flexibility in lower environments.

Azure facilitation

Vulnerability management: Microsoft Defender Vulnerability Management is a comprehensive risk-based vulnerability management solution that you can use to identify, assess, remediate, and track all your biggest vulnerabilities across your most critical assets, all in a single solution.

Next step

[Securely govern your cloud estate](#)

Feedback

Was this page helpful?

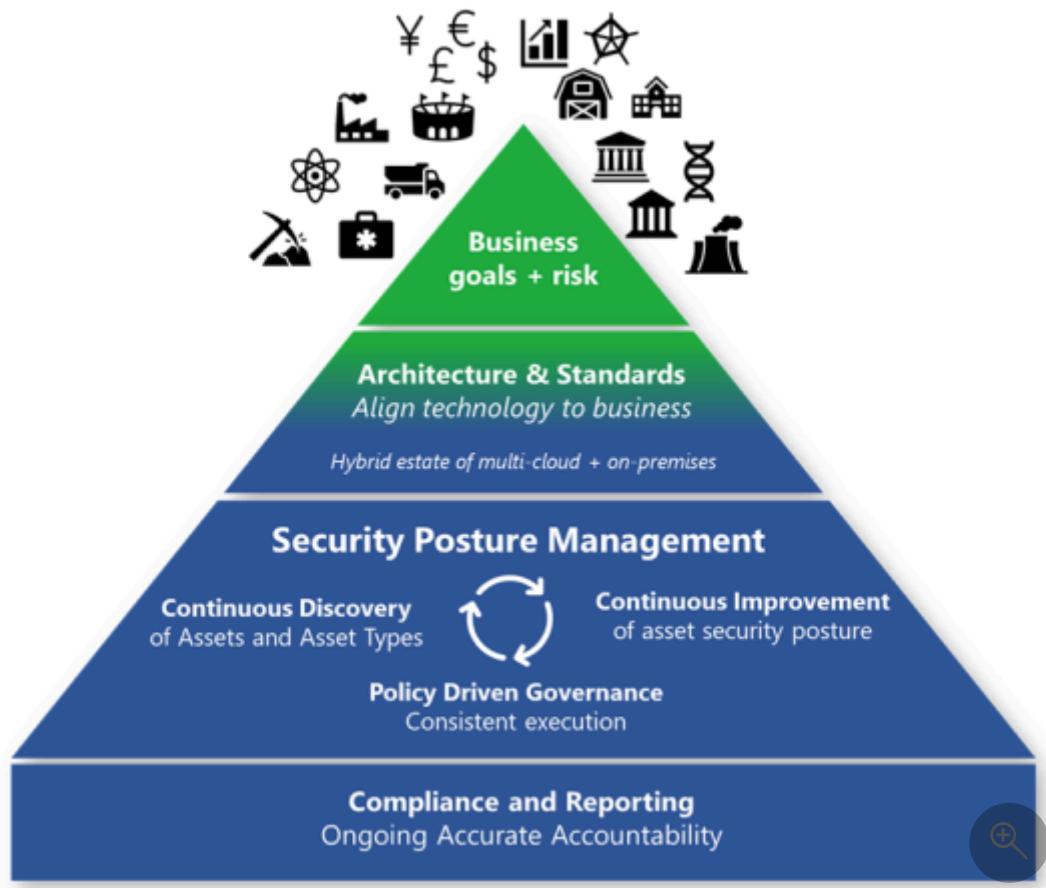
 Yes

 No

Securely govern your cloud estate

Article • 11/18/2024

Security governance connects your business priorities with technical implementations, such as architecture, standards, and policies. Governance teams provide oversight and monitoring to sustain and improve security posture over time. These teams also report compliance that regulatory bodies require.



Business goals and risk provide the most effective guidance for security. This approach ensures that security efforts are concentrated on the organization's key priorities. Additionally, it helps risk owners by using familiar language and processes within the risk management framework.



This article is a supporting guide to the [Govern](#) methodology. It provides areas of security optimization for you to consider as you move through that phase in your journey.

Security posture modernization

Using only problem reporting isn't an effective strategy for maintaining your security posture. In the cloud era, governance requires an active approach that continuously collaborates with other teams. Security posture management is an emerging and essential function. This role addresses the critical question of environmental security. It encompasses key areas such as vulnerability management and security compliance reporting.

In an on-premises environment, security governance relies on the periodic data that's available about the environment. This approach often results in outdated information. Cloud technology revolutionizes this process by providing on-demand visibility into the current security posture and asset coverage. This real-time insight transforms governance into a more dynamic organization. It fosters closer collaboration with other security teams to monitor security standards, provide guidance, and enhance processes.

In its ideal state, governance drives continuous improvement throughout the organization. This ongoing process engages all parts of the organization to ensure constant security advancements.

The following are key principles for security governance:

- **Continuous discovery of assets and asset types:** A static inventory isn't possible in a dynamic cloud environment. Your organization must focus on the continuous discovery of assets and asset types. In the cloud, new types of services are added regularly. Workload owners dynamically adjust the number of application and service instances as needed, which creates a constantly changing environment. This situation makes inventory management a continuously evolving discipline. Governance teams need to continuously identify asset types and instances to keep up with this pace of change.
- **Continuous improvement of asset security posture:** Governance teams should focus on improving and enforcing standards to keep up with the cloud and attackers. Information technology (IT) organizations must react quickly to new threats and adapt accordingly. Attackers constantly evolve their techniques, while defenses continuously improve and might need to be updated. You can't always incorporate all necessary security measures in the initial setup.
- **Policy-driven governance:** This governance ensures consistent implementation because you define policies once and apply them automatically across resources. This process limits wasted time and effort on repeated manual tasks. It's often implemented by using Azure Policy or non-Microsoft policy automation frameworks.

To maintain agility, best practices guidance is often iterative. It digests small pieces of information from multiple sources to create the whole picture and continuously make small adjustments.

Azure facilitation

- [Microsoft Defender for Cloud](#) can help you continuously discover and automatically manage virtual machines in your environment through automatic data collection provisioning.
- [Microsoft Defender for Cloud apps](#) can help you continuously discover and govern first-party and non-Microsoft software as a service apps that are used in your environments.

Incident preparedness and response

Security governance is critical for maintaining your preparedness. To strictly enforce standards, robust governance mechanisms and practices must support the implementation of preparedness and response mechanisms and operational practices. Consider the following recommendations to help govern incident preparedness and response standards:

Incident preparedness governance

- **Automate governance:** Use tooling to automate governance as much as possible. You can use tooling to manage policies for infrastructure deployments, implement hardening measures, protect data, and maintain identity and access management standards. By automating the governance of these security measures, you can ensure that all resources in your environment are compliant with your own security standards and all compliance frameworks that are required for your business. For more information, see [Enforce cloud governance policies](#).
- **Adhere to security baselines from Microsoft:** Understand the security recommendations from Microsoft for the services in your cloud estate, which are available as [security baselines](#). These baselines can help you ensure that your existing deployments are properly secured and that new deployments are correctly configured from the start. This approach reduces the risk of misconfigurations.

Incident response governance

- **Governance of the incident response plan:** The incident response plan should be maintained with the same care as the other critical documents in your estate. Your incident response plan should be:
 - Version controlled to ensure that teams are working off of the most recent version, and that the versioning can be audited.
 - Stored in highly available and secure storage.
 - Reviewed regularly and updated when changes to the environment require it.
- **Governance of incident response training:** Training materials for incident response should be version-controlled for auditability and to ensure that the most recent version is being used at any given time. They should also be reviewed regularly and updated when updates to the incident response plan are made.

Azure facilitation

- [Azure Policy](#) is a policy management solution that you can use to help enforce organizational standards and to assess compliance at-scale. To automate policy enforcement for many Azure services, take advantage of [built-in policy definitions](#).
- [Defender for Cloud](#) provides security policies that can automate compliance with your security standards.

Confidentiality governance

Effective governance is crucial for maintaining security and compliance in enterprise cloud environments. Governance includes the policies, procedures, and controls that ensure data is managed securely and in accordance with regulatory requirements. It provides a framework for decision-making, accountability, and continuous improvement, which is essential for protecting sensitive information and maintaining trust. This framework is crucial for upholding the principle of confidentiality from the CIA Triad. It helps you ensure that sensitive data is accessible only to authorized users and processes.

- **Technical policies:** These policies include access control policies, data encryption policies, and data masking or tokenization policies. The goal of these policies is to create a secure environment by maintaining data confidentiality through stringent access controls and robust encryption methods.
- **Written policies:** Written policies serve as the governing framework for the entire enterprise environment. They establish the requirements and parameters for data

handling, access, and protection. These documents ensure consistency and compliance across the organization and provide clear guidelines for employees and IT staff. Written policies also serve as a reference point for audits and assessments, which helps identify and address any gaps in security practices.

- **Data loss protection:** Continuous monitoring and auditing of data loss prevention (DLP) measures should be conducted to ensure ongoing compliance with confidentiality requirements. This process includes regularly reviewing and updating DLP policies, conducting security assessments, and responding to any incidents that might compromise data confidentiality. Establish DLP programmatically across the organization to ensure a consistent and scalable approach to protecting sensitive data.

Monitor compliance and methods of enforcement

It's critical to monitor compliance and enforce policies to maintain the principle of confidentiality in enterprise cloud environments. These actions are essential for robust security standards. These processes ensure that all security measures are consistently applied and effective to help protect sensitive data from unauthorized access and breaches. Regular assessments, automated monitoring, and comprehensive training programs are essential to ensure adherence to established policies and procedures.

- **Regular audits and assessments:** Conduct regular security audits and assessments to ensure that policies are being followed and identify areas for improvement. These audits should cover regulatory, industry, and organizational standards and requirements, and might involve third-party assessors to provide an unbiased evaluation. An approved assessment and inspection program helps maintain high standards of security and compliance, and ensures that all aspects of data confidentiality are thoroughly reviewed and addressed.
- **Automated compliance monitoring:** Tools like [Azure Policy](#) automate the monitoring of compliance with security policies and provide real-time insights and alerts. This functionality helps ensure continuous adherence to security standards. Automated monitoring helps you detect and respond to policy violations quickly, which reduces the risk of data breaches. It also ensures continuous compliance by regularly checking configurations and access controls against established policies.
- **Training and awareness programs:** Educate employees about data confidentiality policies and best practices to foster a security-conscious culture. Regular training sessions and awareness programs help ensure that all staff members understand

their roles and responsibilities in maintaining data confidentiality. These programs should be updated regularly to reflect changes in policies and emerging threats. This strategy ensures that employees are always equipped with the latest knowledge and skills.

Integrity governance

To maintain your integrity protections effectively, you need a well-designed governance strategy. This strategy should ensure that all policies and procedures are documented and enforced, and that all systems are continuously audited for compliance.

The guidance described previously in the Confidentiality governance section also applies to the principle of integrity. The following recommendations are specific to integrity:

- **Automated data quality governance:** Consider using an off-the-shelf solution to govern your data. Use a prebuilt solution to relieve your data governance team's burden of manual quality validation. This strategy also reduces the risk of unauthorized access and alterations to data during the validation process.
- **Automated system integrity governance:** Consider using a centralized, unified tool to automate your system integrity governance. For example, [Azure Arc](#) allows you to govern systems across multiple clouds, on-premises data centers, and edge sites. By using a system like this, you can simplify your governance responsibilities and reduce operational burden.

Azure facilitation

- [Microsoft Purview Data Quality](#) allows users to assess data quality by using no-code/low-code rules, including out-of-the-box (OOB) rules and AI-generated rules. These rules are applied at the column level and then aggregated to provide scores for data assets, data products, and business domains. This approach ensures comprehensive visibility of data quality across each domain.

Availability governance

The architecture designs that you standardize in your cloud estate require governance to ensure that they aren't deviated from and that your availability isn't compromised by nonconforming design patterns. Likewise, your disaster recovery plans must also be governed to ensure that they're well-maintained.

Availability design governance

- **Maintain standardized design patterns:** Codify and strictly enforce infrastructure and application design patterns. Govern the maintenance of design standards to ensure that they remain up-to-date and protected from unauthorized access or alteration. Treat these standards with the same care as other policies. When possible, automate the enforcement of maintaining design patterns. For example, you can enable policies to control which types of resources can be deployed and specify the regions where deployments are permitted.

Disaster recovery governance

- **Governance of the disaster recovery plans:** Treat disaster recovery plans with the same level of importance as incident response plans. Disaster recovery plans should be:
 - Version-controlled to ensure that teams are always working with the most recent version and that versioning can be audited for accuracy and compliance.
 - Stored in highly available and secure storage.
 - Reviewed regularly and updated when changes to the environment are needed.
- **Governance of disaster recovery drills:** Disaster recovery drills aren't only for training on the plans but also serve as learning opportunities to improve the plan itself. They can also help refine operational or design standards. Meticulous record keeping of disaster recovery drills helps identify areas for improvement and ensures compliance with auditing requirements for disaster preparedness. By storing these records in the same repository as the plans, you can help keep everything organized and secure.

Sustaining secure governance

Modern Service Management (MSM)

Modern Service Management (MSM) is a set of practices and tools designed to manage and optimize IT services in a cloud environment. The goal of MSM is to align IT services with business needs. This approach ensures efficient service delivery while maintaining high standards of security and compliance. MSM provides a structured approach to managing complex cloud environments. MSM also allows organizations to respond quickly to changes, mitigate risks, and ensure continuous improvement. Additionally, MSM is relevant to the principle of confidentiality because it includes tools and practices that enforce data protection and monitor access controls.

- **Unified security management:** MSM tools provide comprehensive security management by integrating various security functions to provide a holistic view of the cloud environment. This approach helps enforce security policies and detects and responds to threats in real-time.
- **Policy management and compliance:** MSM facilitates the creation, enforcement, and monitoring of policies across the cloud environment. It ensures that all resources comply with organizational standards and regulatory requirements. Additionally, it provides real-time insights and alerts.
- **Continuous monitoring and improvement:** MSM emphasizes continuous monitoring of the cloud environment to identify and address potential issues proactively. This approach supports ongoing optimization and improvement of IT services, which ensures that they remain aligned with business objectives.

Next step

Manage your cloud estate with enhanced security

Feedback

Was this page helpful?

 Yes

 No

Manage your cloud estate with enhanced security

Article • 11/18/2024

The Manage phase of a cloud adoption journey focuses on the ongoing operation of your cloud estate. Maintaining and strengthening your security posture continuously is critical to successfully managing your estate and should be considered the cornerstone of your management practices. If you neglect security in favor of cost savings or performance improvements, you risk exposing your business to threats that could severely damage your business and reverse any short-term benefits that doing so brought. Investing in security mechanisms and practices sets your business up for long-term success by minimizing the risks of detrimental attacks.



This article is a supporting guide to the [Manage](#) methodology. It describes areas of security optimization that you should consider as you move through that phase in your journey.

Security posture modernization

During the Manage phase of your cloud adoption journey, you should have a robust [observability platform](#) with thorough monitoring and intelligent alerting set up already, but modernizing this platform might require a new mindset that focuses heavily on proactive measures and adopting the principles of Zero Trust.

- **Assume breach:** Assuming that there's a breach in one or more of your systems is a key tenet of proactive detection and the driver of threat hunting and detection engineering. *Threat hunting* uses a hypothesis-based approach - that a breach has already happened in some particular form - to intelligently analyze your systems through tooling in an attempt to prove or disprove that hypothesis. *Detection engineering* is the practice of developing specialized detection mechanisms to augment observability platforms that aren't equipped to detect new and novel cyberattacks.
- **Verify explicitly:** Moving from a mindset of "trust by default" to "trust by exception" means that you need to be able to validate trusted activities through

visibility. Augmenting your observability platform with intelligent identity and access monitoring can help you detect anomalous behavior in real time.

Azure facilitation

- Microsoft Defender XDR provides advanced threat hunting across multiple domains, like endpoints, cloud apps, and identity.

Managing incident preparedness and response

- **Incident preparedness:**
 - Implement a security information and event management (SIEM) and security orchestration, automation, and response (SOAR) solution to augment your infrastructure monitoring and alerting systems to detect and respond to security incidents.
 - Proactively scan your cloud systems for vulnerabilities. Using a vulnerability scanner that can be integrated with a SIEM system consolidates security data from across your environment, which helps you efficiently detect and respond to multiple types of security risks and incidents.
 - Increase the depth of your visibility into security risks in your environment by implementing an extended detection and response (XDR) solution. Feeding this data into your SIEM system unifies security monitoring into a single pane of glass and optimizes your [security operations](#) team's efficiency.
- **Incident response planning:** Modernizing your observability platform is essential for incident detection. It's also the foundation for maintaining your incident response plan. Your incident response plan must be a living document that's updated regularly. It needs to stay up to date with your threat hunting and detection engineering efforts and with publicly available risk information like the [MITRE ATT&CK](#) knowledge base.

In addition to maintaining your incident response plans, you also need to have fully developed incident response and disaster recovery plans.

- **Business continuity and disaster recovery:** Develop and test disaster recovery plans to ensure that your cloud environment is resilient and can quickly recover from incidents. Include backup and recovery strategies that support business continuity. In many cases, individual workloads in your environment have unique recovery targets and processes, so having workload-based plans, rather than a

single plan that covers all facets of the business, is a good strategy. Refer to the Well-Architected Framework [disaster recovery guide](#) for workload-focused guidance on this topic.

Azure facilitation

- Microsoft Defender for Cloud offers plans that monitor and protect many workload resources, like [servers](#), [storage](#), [containers](#), [SQL databases](#), and [DNS](#). These plans enable you to discover deep insights that you might otherwise be unable to find with your existing monitoring solution.
 - Defender for Servers includes [Microsoft Defender Vulnerability Management](#) for vulnerability scanning against your Azure-based or Azure Arc-enabled VMs.
- Microsoft Sentinel is the Microsoft cloud-native SIEM and SOAR solution. You can use it as a standalone solution. It also integrates with Microsoft Defender to provide a [unified security operations platform](#).
- Automated investigation and response in Defender XDR helps your security operations team address threats more efficiently and effectively by providing automated detection and self-healing capabilities for many scenarios.

Managing confidentiality

The ongoing management of your security posture as it relates to confidentiality involves regularly performing well-designed monitoring and auditing practices, maintaining codified audit procedures, and looking for continuous improvement opportunities.

- **Regular monitoring and auditing:** To ensure the integrity of confidentiality policies, you need to establish a regular cadence for both monitoring and auditing. Continuous monitoring helps with the early detection of potential security threats and anomalies. However, monitoring alone is insufficient. You need to conduct regular audits to verify that the policies and controls in place are effective and that they're being adhered to. Audits provide a comprehensive review of your security posture and help you identify any gaps or weaknesses that you need to address.
- **Documenting and institutionalizing audit procedures:** Documenting audit procedures is crucial for consistency and accountability. Institutionalizing these procedures ensures that audits are conducted systematically and regularly. Detailed documentation should include the scope of the audit, methodologies, tools used, and the frequency of audits. This practice helps you maintain a high

standard of security. It also provides a clear trail for compliance and regulatory purposes.

- **Best practices for enhancing confidentiality include:**

- *Separation of duties (SoD)*: Implementing SoD helps prevent conflicts of interest and reduces the risk of fraud. Dividing responsibilities among different individuals ensures that no single person has control over all aspects of a critical process.
- *Proactive user lifecycle maintenance*: You need to regularly update and manage user accounts. This practice includes promptly revoking access for users who no longer need it, updating permissions as roles change, and ensuring that inactive accounts are disabled. Proactive maintenance helps prevent unauthorized access and helps ensure that only current, authorized users have access to sensitive data. The Access Architects should include these measures in their standard operating procedures.

Azure facilitation

- [Microsoft Purview Data Loss Prevention \(DLP\)](#) can help you detect and prevent exfiltration through common processes used by attackers. Purview DLP can detect adversaries that use any first time use or cloud application to exfiltrate sensitive data from endpoint devices. Purview DLP can also identify the execution of these tools when adversaries rename them to remain undetected.
- [Microsoft Purview Insider Risk Management](#) can help you detect and prevent potential malicious or inadvertent insider risks, such as IP theft, data leakage, and security violations.

Managing integrity

Managing your data and system integrity requires robust monitoring with specific configurations for detecting unauthorized changes to your assets. Other key tenets of the Manage phase are adopting continuous improvement and training practices.

- **Data integrity monitoring**: Effectively monitoring data integrity is a complex task. Intelligent tooling can ease the burden of configuring the appropriate monitoring mechanisms. If you combine intelligent data governance with SIEM and SOAR solutions, you can gain deep insights into activities that are related to your data and automate parts of your incident response plan. Your monitoring should detect anomalous behaviors, including unauthorized access to data stores and changes to

data stores. Automated incident responses like immediate lockouts can help minimize the blast radius of malicious activities.

- **System integrity monitoring:** Effectively monitoring for system integrity is more straightforward than monitoring data integrity. Most modern monitoring and alerting platforms are well equipped to detect changes to systems. With proper guardrails around deployments, like only allowing changes to the environment through IaC, and a well-designed authentication and access platform, you can ensure that changes that occur outside of the approved protocols are detected and investigated immediately.

Azure facilitation

Data integrity monitoring

- [Microsoft Purview health management](#) can help you codify data standards and measure how the data in your estate complies with those standards over time. It provides reports to track data health and helps data owners remediate issues that arise.

Managing availability

Managing the availability of your cloud estate requires robust, proactive availability monitoring that's validated through testing.

- **Availability monitoring:** Ensure that all infrastructure and applications are configured for monitoring and that alerting is configured to notify the appropriate teams. Use cloud-native logging and [application instrumenting](#) functionality to simplify your monitoring design and reduce operational burden.
- **Availability testing:** All infrastructure and applications must be tested regularly for availability as part of your overall testing strategy. [Fault injection and chaos testing](#) are excellent strategies for testing availability and security by purposely introducing malfunctions.

Azure facilitation

In addition to the Defender for Cloud solutions discussed previously, consider the following solutions:

- [Automatic instrumentation for Azure Monitor Application Insights](#) allows you to easily instrument your application for rich telemetry monitoring through [Application Insights](#). Many Azure-based and on-premises hosting types are supported for automatic instrumentation.
- [Azure Chaos Studio](#) is a managed service that uses chaos engineering to help you measure, understand, and improve your cloud application and service resilience.

Managing security sustainment

Continuous education

Encourage ongoing education and certification in cloud security practices to keep up to date with evolving threats and technologies. This training should cover:

- **Threat detection.** Use advanced analytics and monitoring tools like Microsoft Sentinel to detect threats early, emphasizing continuous monitoring and proactive identification of threats. Advanced analytics enable the identification of unusual patterns and behaviors that might indicate potential security threats. Integrated threat intelligence provides up-to-date information on known threats, which enhances the system's ability to detect emerging risks. Include training on preplanned responses, such as automated actions for containment, to ensure rapid reaction to detected threats.
- **Incident response.** Train your security operations team on robust incident response strategies that integrate Zero Trust principles, assuming threats can come from both internal and external sources. This activity includes continuous verification of identities and securing access to resources. Training should also cover the use of decision trees and flowcharts to guide response actions based on specific incident scenarios.
 - **Availability.** Provide training on deploying high availability and disaster recovery solutions by using Azure services to ensure that data and resources remain accessible when they're needed. This training includes maintaining preplanned responses that outline the steps for preserving availability during an incident. Training should also cover strategies for ensuring continuous access to critical resources, even in the face of disruptions, and include hands-on training on setting up and managing Azure high availability and disaster recovery tools.
- **Simulation exercises:** Conduct regular security drills and simulations to prepare the team for real-world scenarios. These exercises should evaluate the organization's ability to respond to incidents within the Zero Trust framework,

treating all network segments as potentially compromised until they're verified as secure. Scenarios such as phishing attacks, data breaches, and ransomware should be simulated to identify gaps in response strategies and provide hands-on experience with handling incidents. Drills should emphasize containment strategies by quickly isolating compromised systems to prevent further spread, rapid communication through the establishment of clear and efficient channels for disseminating information, and evidence preservation by ensuring that all relevant data is securely collected and stored to support subsequent analysis and investigation. Use preplanned responses like incident playbooks and communication protocols to ensure that actions during these drills are consistent and systematic.

- **Incident response drills:** Regularly test incident response plans through realistic drills that simulate various threat scenarios. These drills should involve all relevant stakeholders, including the Security Operations Center (SOC) team, incident response coordinators, governance lead, incident controller, investigation lead, infrastructure lead, and cloud governance team, to ensure comprehensive preparedness across the organization. Incorporate elements of the CIA triad and Zero Trust principles into these drills, such as testing the effectiveness of access controls (confidentiality), implementing integrity checks for critical data, and implementing procedures for maintaining service availability during an incident. Emphasize effective coordination by ensuring clear communication and collaborative efforts across teams through the use of preplanned responses, like predefined roles and responsibilities, and rapid containment through swift isolation of affected systems and threat mitigation. Document actions taken to provide a clear record for post-incident review and continuous improvement.

Continuous improvement strategies for confidentiality and integrity

Continuous improvement is essential for maintaining and enhancing confidentiality and integrity in enterprise cloud environments. Documenting the results of configuration management and audits and inspections is crucial. This documentation provides a historical record that you can analyze to identify trends, recurring issues, and areas for improvement.

- **Confidentiality strategies:**
 - *Learn from the past.* Implementing lessons learned from past inspections is a key tenet of continuous improvement. By analyzing the outcomes of previous audits and inspections, organizations can identify weaknesses and implement

corrective actions to prevent similar issues in the future. This proactive approach ensures that the organization continuously evolves and improves its security posture.

- *Use real-time data.* Real-time monitoring plays a critical role in continuous improvement. By using real-time data, organizations can quickly identify and respond to potential threats, which ensures that security measures are always up to date and effective. This dynamic approach helps organizations avoid repeating past mistakes and ensures that organizations remain resilient against emerging threats.
- *Confidentiality training.* As part of your overall training strategy, ensure that employees are trained on your confidentiality policies and procedures. This training should be mandatory for new hires and should recur regularly for all employees. For employees on the security team, it should include deeper training that's specific to their roles. Teach the importance of implementing encryption and strict access controls to protect sensitive information from unauthorized access. Training should also cover best practices in data encryption technologies and access management tools to help ensure that only authorized personnel can access sensitive data.

- **Integrity strategies:**

- *Routinely audit your data.* Routinely perform manual audits of your data to ensure that your data governance and monitoring tooling is performing as expected. Look for opportunities for improvement.
- *Data hygiene.* Adopt good data hygiene habits, like the following.
 - Manually audit data for quality, accuracy, and consistency. Correct errors when they're discovered.
 - Use strategies like normalization to reduce inconsistencies and redundancies.
 - Archive historical data in cold or offline storage when it's no longer needed in production. Purge data that doesn't need to be archived.
 - Regularly review encryption configurations to ensure that all sensitive data is encrypted at rest and in transit. Regularly review industry standards for encryption and ensure that your systems align with those standards.
- *Backup.* Regularly review backup configurations to ensure that all data stores that contain sensitive data or other critical data are being backed up. Perform

restore tests to ensure that the backup data is valid. Regularly test restores to ensure that your systems are in compliance with your organization's recovery time objective (RTO) and recovery point objective (RPO) targets.

- *Regularly review access to systems and data.* Reviews of access permissions to systems and data stores should happen regularly to ensure that there are no gaps in your access controls and policies.
- *Conduct integrity training.* As part of your overall training strategy, ensure that employees are trained on your data and system integrity policies and procedures. This training should be mandatory for new hires and recur regularly for all employees. For employees on the security team, provide deeper training that's specific to their roles. Provide training on the use of DevOps processes for infrastructure as code (IaC) to help ensure data accuracy and reliability. DevOps practices, such as version control, continuous integration, and automated testing, help you track, audit, and validate changes to the cloud environment's infrastructure before deployment. DevOps practices are particularly important for maintaining landing zones, because these practices ensure consistency and integrity in the configuration by providing a systematic way to handle infrastructure changes.

Next step

Review the [Zero Trust adoption framework](#) to learn about integrating Zero Trust approaches throughout your cloud adoption journey.

Review the Well-Architected Framework [Security](#) pillar to get workload-focused security guidance.

Feedback

Was this page helpful?

 Yes

 No

Manage organizational alignment

Article • 02/28/2023

Cloud adoption can't happen without well-organized people. Successful cloud adoption comes from skilled people doing appropriate types of work. This work is in alignment with clearly defined business goals in a well-managed environment. To deliver an effective operating model for the cloud, it's important to establish appropriately staffed organizational structures. This article outlines an approach to establishing and maintaining the proper organizational structures in four steps.

The following exercises will help guide the process of creating a landing zone to support cloud adoption.

- 1 **Structure type:** Define the type of organizational structure that best fits your operating model.
- 2 **Cloud functions:** Understand the cloud functionality required to adopt and operate the cloud.
- 3 **Mature team structures:** Define the teams that can provide various cloud functions.
- 4 **RACI matrix:** Use the provided RACI matrix to map roles to each team for functions of the cloud operating model. This matrix includes responsibility, accountability, consulted, and informed roles.

Structure type

The following organizational structures don't necessarily have to map to an organizational chart, or org chart. Org charts generally reflect command and control management structures. Conversely, the following organizational structures are designed to capture alignment of roles and responsibilities. In an agile, matrix organization, these structures might be best represented as virtual teams. There's no limitation suggesting that these organizational structures can't be represented in an org chart. However, it's not necessary to produce an effective operating model.

The first step of managing organizational alignment is to determine how the following organizational structures will be fulfilled:

- **Org chart alignment:** Management hierarchies, manager responsibilities, and staff alignment will align to organizational structures.
- **Virtual teams:** Management structures and org charts remain unchanged. Instead, virtual teams will be created and tasked with the required functions.
- **Mixed model:** More commonly, a mixture of org chart and virtual team alignment will be required to deliver on transformation goals.

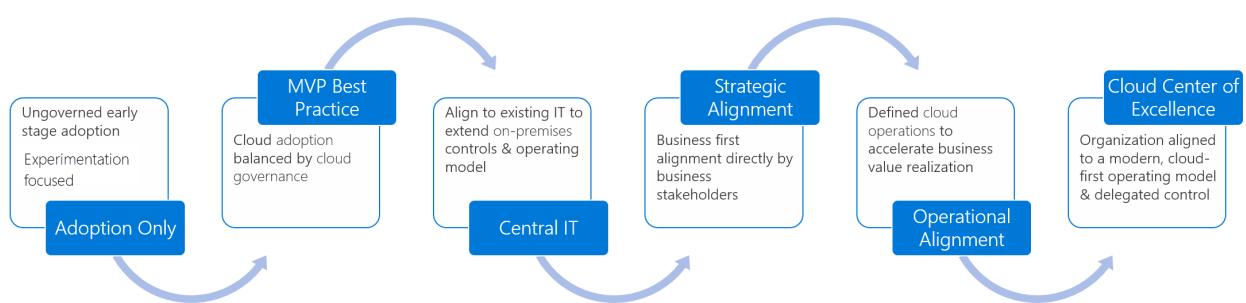
Understand required cloud functions

This list contains the functions required to succeed at cloud adoption and longer-term operating models. After you're familiar with these functions, you can align them to organizational structures based on staffing and maturity.

- **Cloud strategy:** Align technical change to business needs.
- **Cloud adoption:** Deliver technical solutions.
- **Cloud governance:** Manage risk.
- **Central IT team:** Support from existing IT staff.
- **Cloud operations:** Support and operate adopted solutions.
- **Cloud center of excellence:** Improve quality, speed, and resiliency of adoption.
- **Cloud platform:** Operate and mature the platform.
- **Cloud automation:** Accelerate adoption and innovation.
- **Cloud data:** Manage data and enable analytics solutions.
- **Cloud security:** Manage information security risk.

To some degree, each of these functions are delivered in every cloud adoption effort. The delivery is either explicit or follows a defined team structure.

As adoption needs grow, so does the need to create balance and structure. To meet those needs, companies often follow a process of maturing organizational structures.



The article on [determining organizational structure maturity](#) provides more details about each level of maturity.

To track organization structure decisions over time, download and modify the [RACI template](#).

Understand the function of cloud strategy

Article • 02/28/2023

A cloud strategy team defines motivations and business outcomes for cloud adoption. They then validate and maintain alignment between business priorities and cloud adoption efforts. Your company ideally forms a strategy that clearly ties technical activities to the business outcomes. In the absence of a defined cloud strategy team, someone must still provide those functions. That same person or group then also manages changes across the project.

Typically, the following roles provide cloud strategy functions. When you define your cloud strategy team, be sure it includes many of the following roles:

- Finance
- Line of business
- Human resources
- Operations
- Enterprise architecture
- IT infrastructure
- Application groups
- Project managers (often with agile project management experience)

These roles help guide critical prioritization and discovery efforts during cloud adoption. This exploration might also trigger changes in business processes, operations, customer interactions, or even product development. When these functions are confined to IT, the success of cloud adoption efforts is often constrained. To drive true business change, business leaders should be the primary champions of this functionality. A defined cloud strategy team provides a means for involving key participants in a structured way.

Note

The organization's CEO and CIO often assign the team. Assignments are typically based on empowering the cloud strategy team to drive changes that span other organizations within the enterprise. Typically, business leaders assign cloud strategy team members based on the **motivations for cloud adoption**, **business outcomes**, and relevant **financial models**.

Preparation

- Learn the business value of Microsoft Azure.
- Learn how the [Cloud Adoption Framework](#) can help you align the strategy for business, people, and technology.
- Review the [cloud adoption strategy](#) process.
- Download the [strategy and plan template](#).

Minimum scope

Align business stakeholders to maximize the business value of cloud adoption investments.

Whenever possible, define the business outcomes and cloud strategy early in the process. As investments in cloud adoption grow and the team sees the business values, stakeholders often become more engaged. When cloud adoption efforts are led by the business, the focus might be on an operating model and the organization.

Establish a vision

- [Adoption motivations](#): Document and articulate the reasons behind the technical effort.
- [Business outcomes](#): Clearly articulate what's expected of the technical team in terms of business changes.
- [Learning metrics](#): Establish short-term metrics that show progress toward longer-term business outcomes.

Build business justification

- [Cloud migration business case](#): Establish a business case for cloud migration.

Rationalize the digital estate

- [Incremental rationalization](#): An agile approach to rationalization that properly aligns late-bound technical decisions.
- [The five Rs of rationalization](#): Understand the various rationalization options.

Deliverable

The cloud strategy team drives critical prioritization and discovery efforts during cloud adoption. They might also change business processes, operations, customer interactions, or even product development. The primary focus of the cloud strategy

team is to validate and maintain alignment between business priorities and cloud adoption efforts. Secondarily, this team focuses on change management across the adoption efforts. Ideally, the cloud strategy team delivers on the following tasks and allocates sufficient time for planning.

Early planning tasks:

- Review and provide feedback on business outcomes and financial models
- Help establish clear motivations for cloud adoption that align with corporate objectives
- Define relevant learning metrics that clearly communicate progress toward business outcomes
- Understand business risks introduced by the plan that represent the business's tolerance for risk
- Review and approve the rationalization of the digital estate

Ongoing monthly tasks:

- Support the cloud governance team during risk and tolerance conversations
- Review release plans to understand timelines and business outcomes of the technical change
- Define business change plans associated with planned releases
- Ensure business teams are ready to run business testing and the business change plan

Meeting cadence:

Cloud strategy team members must allocate time to plan and develop the cloud strategy:

- During early planning efforts, allocate an hour each week to meet with the team. After the team solidifies the adoption plan (usually within 4-6 weeks), the time requirements can be reduced.
- Throughout the adoption efforts, allocate 1-2 hours each month to review progress and validate continued priorities.
- More time is likely required from delegated members of the executive's team on an as-needed basis. Each member of the cloud strategy team should appoint a delegate who can allocate 5-10 hours per week to support ongoing prioritization questions and report on any urgent needs.

Next steps

- Start a [cloud strategy team](#).

- Align your strategy with the [cloud adoption functions](#) by creating a [cloud adoption team](#).
- Use the [RACI template](#) to align responsibilities across teams.

Cloud adoption functions

Article • 02/28/2023

Cloud adoption functions enable the implementation of technical solutions in the cloud. Like any IT project, the people delivering the actual work will determine success. The teams providing the necessary cloud adoption functions can be staffed from multiple subject matter experts or implementation partners.

Cloud adoption teams are the modern-day equivalent of technical implementation teams or project teams. But the nature of the cloud may require a more fluid team structure. Some teams focus exclusively on cloud migration, while other teams focus on innovations that take advantage of cloud technologies. Some teams include the broad technical expertise required to complete large adoption efforts, like a full datacenter migration. Other teams have a tighter technical focus and may move between projects to accomplish specific goals. One example would be a team of data platform specialists who help convert SQL VMs to SQL PaaS instances.

Regardless of the type or number of cloud adoption teams, the functionality required for cloud adoption is provided by subject matter experts found in IT, business analysis, or implementation partners.

Depending on the desired business outcomes, the skills needed to provide full cloud adoption functions could include:

- Infrastructure implementers
- DevOps engineers
- Application developers
- Data scientists
- Data or application platform specialists

For optimal collaboration and efficiency, we recommend that cloud adoption teams have an average team size of six people. These teams should be self-organizing from a technical execution perspective. We highly recommend that these teams also include project management expertise, with deep experience in agile, Scrum, or other iterative models. This team is most effective when managed using a flat structure.

Preparation

- [Create an Azure account](#): The first step to using Azure is to create an account.
- [Azure portal](#): Tour the Azure portal features and services, and customize the portal.

- [Introduction to Azure](#): Get started with Azure. Create and configure your first virtual machine in the cloud.
- [Azure fundamentals](#): Learn cloud concepts, understand the benefits, compare and contrast basic strategies, and explore the breadth of services available in Azure.
- Review the [Migrate methodology](#).

Minimum scope

The nucleus of all cloud adoption efforts is the cloud migration team. This team drives the technical changes that enable adoption. Depending on the objectives of the adoption effort, this team may include a diverse range of team members who handle a broad set of technical and business tasks.

At a minimum, the team scope includes:

- [Rationalization of the digital estate](#)
- Review, validation, and advancement of the [prioritized migration backlog](#)
- The execution of the [first workload](#) as a learning opportunity.

Deliverable

The primary need from any cloud adoption function is the timely, high-quality implementation of the technical solutions outlined in the adoption plan. These solutions should align with governance requirements and business outcomes, and should take advantage of technology, tools, and automation solutions that are available to the team.

Early planning tasks:

- Execute the [rationalization of the digital estate](#).
- Review, validate, and advance the [prioritized migration backlog](#).
- Begin execution of the [first workload](#) as a learning opportunity.

Ongoing monthly tasks:

- Oversee [change management processes](#).
- Manage the [release and sprint backlogs](#).
- Build and maintain the adoption landing zone in conjunction with governance requirements.
- Execute the technical tasks outlined in the [sprint backlogs](#).

Meeting cadence:

We recommend that teams providing cloud adoption functions be dedicated to the effort full-time.

It's best if these teams meet daily in a self-organizing way. The goal of daily meetings is to quickly update the backlog, and to communicate what has been completed, what is to be done today, and what things are blocked, requiring additional external support.

Release schedules and iteration durations are unique to each company. But a range of one to four weeks per iteration seems to be the average duration. Regardless of iteration or release cadence, we recommend that the team meets all supporting teams at the end of each release to communicate the outcome of the release, and to reprioritize upcoming efforts. Likewise, it's valuable to meet as a team at the end of each sprint, with the cloud center of excellence or cloud governance team to stay aligned on common efforts and any needs for support.

Some of the technical tasks associated with cloud adoption can become repetitive. Team members should rotate every 3–6 months to avoid employee satisfaction issues and maintain relevant skills. A rotating seat on a cloud center of excellence or cloud governance team can provide an excellent opportunity to keep employees fresh and harness new innovations.

Learn more about the function of a [cloud center of excellence](#) or [cloud governance team](#).

Next steps

- [Build a cloud adoption team](#)
- Align cloud adoption efforts with [cloud governance functions](#) to accelerate adoption and implement best practices, while reducing business and technical risks.

Cloud governance functions

Article • 02/28/2023

A cloud governance team ensure that risks and risk tolerance are properly evaluated and managed. This team ensures the proper identification of risks that can't be tolerated by the business. The people on this team convert risks into governing corporate policies.

Depending on the desired business outcomes, the skills needed to provide full cloud governance functions include:

- IT governance
- Enterprise architecture
- Security
- IT operations
- IT infrastructure
- Networking
- Identity
- Virtualization
- Business continuity and disaster recovery
- Application owners within IT
- Finance owners

These baseline functions help you identify risks related to current and future releases. These efforts help you evaluate risk, understand the potential impacts, and make decisions regarding risk tolerance. When doing so, quickly update plans to reflect the changing needs of the [cloud migration team](#).

Preparation

- Review the [Govern methodology](#).
- Take the [governance benchmark assessment](#).
- **Introduction to security in Azure:** Learn the basic concepts to protect your infrastructure and data in the cloud. Understand what responsibilities are yours and what Azure handles for you.
- Understand how to work across groups to [manage cost](#).

Minimum scope

- Understand [business risks](#) introduced by the plan.
- Represent the [business's tolerance for risk](#).

- Help create a [governance MVP](#).

Involve the following participants in cloud governance activities:

- Leaders from middle management and direct contributors in key roles should represent the business and help evaluate risk tolerances.
- The cloud governance functions are delivered by an extension of the [cloud strategy team](#). Just as the CIO and business leaders are expected to participate in cloud strategy functions, their direct reports are expected to participate in cloud governance activities.
- Business employees that are members of the business unit who work closely with the leadership of the line-of-business should be empowered to make decisions regarding corporate and technical risk.
- Information technology (IT) and information security (IS) employees who understand the technical aspects of the cloud transformation may serve in a rotating capacity instead of being a consistent provider of cloud governance functions.

Deliverable

The cloud governance mission is to balance competing forces of transformation and risk mitigation. Additionally, cloud governance ensures that the [cloud migration team](#) is aware of data and asset classification, as well as architecture guidelines that govern adoption. Governance teams or individuals also works with the [cloud center of excellence](#) to apply automated approaches to governing cloud environments.

Ongoing monthly tasks:

- Understand [business risks](#) introduced during each release.
- Represent the [business's tolerance for risk](#).
- Aid in the incremental improvement of [policy and compliance requirements](#).

Meeting cadence:

The time commitment from each team member of the cloud governance team will represent a large percentage of their daily schedules. Contributions will not be limited to meetings and feedback cycles.

Out of scope

As adoption scales, the cloud governance team may struggle to keep pace with innovations. This is especially true if your environment has heavy compliance,

operations, or security requirements. If this happens you can shift some responsibilities to an existing IT team to reduce scope for the governance team.

Next steps

Some large organizations have dedicated teams that focus on IT governance. These teams specialize in risk management across the IT portfolio. When those teams exist, the following maturity models can be accelerated quickly. But the IT governance team is encouraged to review the cloud governance model to understand how governance shifts slightly in the cloud. Key articles include extending corporate policy to the cloud and the Five Disciplines of Cloud Governance.

No governance: Organizations often move into the cloud with no clear plans for governance. Before long, concerns around security, cost, scale, and operations begin to trigger conversations about the need for a governance model and people to staff the processes associated with that model. Starting those conversations before they become concerns is always a good first step to overcome the antipattern of *no governance*. The section on defining corporate policy can help facilitate those conversations.

Governance blocked: When concerns around security, cost, scale, and operations go unanswered, projects and business goals tend to get blocked. Lack of proper governance generates fear, uncertainty, and doubt among stakeholders and engineers. Stop this in its tracks by taking action early. The two governance guides defined in the Cloud Adoption Framework can help you start small, set initially limiting policies to minimize uncertainty and mature governance over time. Choose from the complex enterprise guide or standard enterprise guide.

Voluntary governance: There tend to be brave souls in every enterprise. Those gallant few who are willing to jump in and help the team learn from their mistakes. Often this is how governance starts, especially in smaller companies. These brave souls volunteer time to fix some issues and push cloud adoption teams toward a consistent well-managed set of best practices.

The efforts of these individuals are much better than "no governance" or "governance blocked" scenarios. While their efforts should be commended, this approach should not be confused with governance. Proper governance requires more than sporadic support to drive consistency, which is the goal of any good governance approach. The guidance in the Five Disciplines of Cloud Governance can help develop this discipline.

Cloud custodian: This moniker has become a badge of honor for many cloud architects who specialize in early stage governance. When governance practices first start out, the results appear similar to those of governance volunteers. But there is one fundamental

difference. A cloud custodian has a plan in mind. At this stage of maturity, the team is spending time cleaning up the messes made by the cloud architects who came before them. But the cloud custodian aligns that effort to well structured corporate policy. They also use governance tools, like those outlined in the governance MVP.

Another fundamental difference between a cloud custodian and a governance volunteer is leadership support. The volunteer puts in extra hours above regular expectations because of their quest to learn and do. The cloud custodian gets support from leadership to reduce their daily duties to ensure regular allocations of time can be invested in improving cloud governance.

Cloud guardian: As governance practices solidify and become accepted by cloud adoption teams, the role of cloud architects who specialize in governance changes a bit, as does the role of the cloud governance team. Generally, the more mature practices gain the attention of other subject matter experts who can help strengthen the protections provided by governance implementations.

While the difference is subtle, it is an important distinction when building a governance-focused IT culture. A cloud custodian cleans up the messes made by innovative cloud architects, and the two roles have natural friction and opposing objectives. A cloud guardian helps keep the cloud safe, so other cloud architects can move more quickly with fewer messes.

Cloud guardians begin using more advanced governance approaches to accelerate platform deployment and help teams self-service their environmental needs, so they can move faster. Examples of these more advanced functions are seen in the incremental improvements to the governance MVP, such as improvement of the security baseline.

Cloud accelerators: Cloud guardians and cloud custodians naturally harvest scripts and governance tools that accelerate the deployment of environments, platforms, or even components of various applications. Curating and sharing these scripts in addition to centralized governance responsibilities develops a high degree of respect for these architects throughout IT.

Those governance practitioners who openly share their curated scripts help deliver technology projects faster and embed governance into the architecture of the workloads. This workload influence and support of good design patterns elevate cloud accelerators to a higher rank of governance specialist.

Global governance: When organizations depend on globally dispersed IT needs, there can be significant deviations in operations and governance in various geographies. Business unit demands and even local data sovereignty requirements can cause governance best practices to interfere with required operations. In these scenarios, a

tiered governance model allows for minimally viable consistency and localized governance. The article on multiple layers of governance provides more insights on reaching this level of maturity.

Every company is unique, and so are their governance needs. Choose the level of maturity that fits your organization and use the Cloud Adoption Framework to guide the practices, processes, and tooling to help you get there.

As cloud governance matures, teams are empowered to adopt the cloud at faster paces. Continued cloud adoption efforts tend to trigger maturity in IT operations. Either develop a cloud operations team, or sync with your cloud operations team to ensure governance is a part of operations development.

Learn more about starting a [cloud governance team](#) or a [cloud operations team](#).

After you've established an [initial cloud governance foundation](#), use these best practices in [Governance foundation improvements](#) to get ahead of your adoption plan and prevent risks.

Understand the functions of a central IT team

Article • 05/07/2024

As cloud adoption scales, cloud governance functions alone might not be sufficient to govern adoption efforts. When adoption is gradual, teams tend to organically develop the necessary skills and processes to be ready for the cloud over time.

But when one cloud adoption team uses the cloud to achieve a high-profile business outcome, gradual adoption is seldom the case. Success follows success. This result is also true for cloud adoption, but it happens at cloud scale. When cloud adoption expands from one team to multiple teams relatively quickly, the organization needs more support from existing IT staff. But those staff members might lack the training and experience required to support the cloud using cloud-native IT tools. This gap in training and experience often drives the formation of a central IT team to govern the cloud.

⊗ Caution

While setting up a central IT team is a common signal of maturity, if not managed effectively, it can become a high risk to adoption, potentially blocking innovation and migration efforts. See the following [Central IT team risks](#) section to learn how to mitigate the risk of centralization becoming a cultural antipattern.

The following disciplines and structures cover the requirements for setting up centralized IT functions:

- An existing central IT team
- Enterprise architects
- IT operations
- IT governance
- IT infrastructure
- Networking
- Identity
- Virtualization
- Business continuity and disaster recovery
- Application owners within IT

⚠ Warning

You should only apply centralized IT in the cloud when you've based your existing delivery on-premises on the central IT team model. If you based your current on-premises model on delegated control, consider a cloud center of excellence (CCoE) approach for a more cloud-compatible alternative.

Key responsibilities

Adapt existing IT practices to ensure that adoption efforts result in well-governed, well-managed environments in the cloud.

Typically, your team does the following tasks regularly:

Strategic tasks

- Review:
 - [Business outcomes](#)
 - [Financial models](#)
 - [Motivations for cloud adoption](#)
 - [Business risks](#)
 - [Rationalize the digital estate](#)
- Monitor adoption plans and progress against the [prioritized migration backlog](#).
- Identify and prioritize required platform changes that support the migration backlog.
- Act as an intermediary or translation layer between cloud adoption needs and existing IT teams.
- Take advantage of existing IT teams to accelerate platform functions and enable adoption.

Technical tasks

- Build and maintain the cloud platform to support solutions.
- Define and implement the platform architecture.
- Operate and manage the cloud platform.
- Continuously improve the platform.
- Keep up with new innovations in the cloud platform.
- Deliver new cloud functionality to support business value creation.
- Suggest self-service solutions.
- Ensure that solutions meet existing governance and compliance requirements.
- Create and validate deployment of platform architecture.
- Review release plans for sources of new platform requirements.

Meeting cadence

Central IT team expertise usually comes from a working team. Expect participants to commit much of their daily schedules to alignment efforts. Contributions aren't limited to meetings and feedback cycles.

Central IT team risks

Prefix each of the cloud functions and phases of organizational maturity with the word "cloud". The central IT team is the only exception. Centralized IT became prevalent when all IT assets could be housed in few locations, managed by a few teams, and controlled through a single operations management platform. Global business practices and the digital economy have largely reduced the instances of those centrally managed environments.

In the modern view of IT, assets are globally distributed. Responsibilities are delegated. A mixture of internal staff, managed service providers, and cloud providers deliver operations management. In the digital economy, IT management practices are transitioning to a model of self-service and delegated control with clear guardrails to enforce governance. A central IT team can be a valuable contributor to cloud adoption by becoming a cloud broker and a partner for innovation and business agility.

A central IT team is well positioned to take valuable knowledge and practices from existing on-premises models and apply those practices to cloud delivery. But this process requires change. It requires new processes, new skills, and new tools to support cloud adoption at scale. When a central IT team adapts, it becomes an important partner in cloud adoption efforts. But if the central IT team doesn't adapt to the cloud, or attempts to use the cloud as a catalyst for tight-grain controls, it quickly becomes a blocker to adoption, innovation, and migration.

The measures of this risk are speed and flexibility. The cloud simplifies adopting new technologies quickly. When new cloud functionality can be deployed within minutes, but the reviews by the central IT team add weeks or months to the deployment process, these centralized processes become a major impediment to business success. When you encounter this indicator, consider alternative strategies to IT delivery.

Exceptions

Many industries require rigid adherence to third-party compliance. Some compliance requirements still demand centralized IT control. Delivering on these compliance measures can add time to deployment processes, especially for new technologies that

haven't been used broadly. In these scenarios, expect delays in deployment during the early stages of adoption. Similar situations might exist for companies that deal with sensitive customer data, but might not be governed by a third-party compliance requirement.

Operate within the exceptions

When centralized IT processes are required and those processes create appropriate checkpoints in adoption of new technologies, these innovation checkpoints can still be addressed quickly. Governance and compliance requirements are designed to protect those things that are sensitive, not to protect everything. The cloud provides simple mechanisms for acquiring and deploying isolated resources while maintaining proper guardrails.

A mature central IT team maintains necessary protections but negotiates practices that still enable innovation. Demonstrating this level of maturity depends on proper classification and isolation of resources.

Example narrative of operating within exceptions to empower adoption

This example narrative illustrates the approach taken by a mature central IT team at the fictional company Contoso to empower adoption.

Contoso adopts a central IT team model to support the business's cloud resources. To deliver this model, they implement tight controls for various shared services such as ingress network connections. This wise move reduces the exposure of their cloud environment and provides a single "break-glass" device to block all traffic if a breach occurs. Their Security Baseline policies state that all ingress traffic must come through a shared device managed by the central IT team.

But one of their cloud adoption teams now requires an environment with a dedicated and specially configured ingress network connection to use a specific cloud technology. An immature central IT team would just refuse the request and prioritize its existing processes over adoption needs. Contoso's central IT team is different. They quickly identify a simple four-part solution to this dilemma:

1. **Classification:** Because the cloud adoption team is in the early stages of building a new solution and doesn't have any sensitive data or mission-critical support needs, they classify the assets in the environment as low risk and noncritical. Effective classification shows maturity in a central IT team. Classifying all assets and environments allows for clearer policies.

2. **Negotiation:** Classification alone isn't sufficient. The company implements shared services to consistently operate sensitive and mission-critical assets. Changing the rules compromises governance and compliance policies designed for the assets that need more protection. Empowering adoption can't happen at the cost of stability, security, or governance. This leads to a negotiation with the adoption team to answer specific questions. Can a business-led DevOps team provide operations management for this environment? Does this solution require direct access to other internal resources? If the cloud adoption team is comfortable with the tradeoffs, the ingress traffic might be possible.
3. **Isolation:** Because the business provides its own ongoing operations management, and because the solution doesn't rely on direct traffic to other internal assets, the solution is then cordoned off in a new subscription. That subscription is also added to a separate node of the new management group hierarchy.
4. **Automation:** Automation principles are another sign of maturity for this team. The team uses Azure Policy to automate policy enforcement. They also use infrastructure as code (IaC) to automate deployment of common platform components and enforce adherence to the defined identity baseline. The policies and templates vary slightly for this subscription and for all others in the new management group. Policies that block ingress bandwidth are lifted. The policies are then replaced by requirements to route traffic through the shared services subscription, like ingress traffic, to enforce traffic isolation. Because the on-premises operations management tooling can't access this subscription, agents for that tool are also no longer required. All other governance guardrails required by other subscriptions in the management group hierarchy are still enforced, which ensures sufficient guardrails.

The mature creative approach of Contoso's central IT team provides a solution that doesn't compromise governance or compliance, but still encourages adoption. This approach of brokering rather than owning cloud-native approaches to centralized IT is the first step toward building a cloud center of excellence (CCoE). Adopting this approach to quickly evolve existing policies allows for centralized control when it's required and governance guardrails when more flexibility is acceptable. Balancing these two considerations mitigates the risks associated with centralized IT in the cloud.

Next steps

- As a central IT team matures its cloud capabilities, the next maturity step is typically a looser coupling of cloud operations. The availability of cloud-native operations management tooling and lower operating costs for PaaS-first solutions

often lead to business teams (or more specifically, DevOps teams within the business) assuming responsibility for cloud operations.

Learn more about:

- [Building a cloud operations team](#)
 - [Cloud operations functions](#)
-

Feedback

Was this page helpful?

 Yes

 No

Cloud operations functions

Article • 02/28/2023

An operations team focuses on monitoring, repairing, and the remediation of issues related to traditional IT operations and assets. In the cloud, many of the capital costs and operations activities are transferred to the cloud provider, giving IT operations the opportunity to improve and provide significant additional value.

The skills needed to provide cloud operations functions can be provided by:

- IT operations
- Outsource IT operations vendors
- Cloud service providers
- Cloud-managed service providers
- Application-specific operations teams
- Business application operations teams
- DevOps teams

Important

The individuals or teams accountable for cloud operations are generally responsible for making reactive changes to configuration during remediation. They're also likely to be responsible for proactive configuration changes to minimize operational disruptions. Depending on the organization's cloud operating model, those changes could be delivered via infrastructure-as-code, Azure Pipelines, or direct configuration in the portal. Since operations team will likely have elevated permissions, it is extremely important that those who fill this role are following **identity and access control best practices** to minimize unintended access or production changes.

Preparation

- [Manage resources in Azure](#): Learn how to work through the Azure CLI and web portal to create, manage, and control cloud-based resources.
- [Azure network services](#): Learn Azure networking basics and how to improve resiliency and reduce latency.

Review the following:

- [Business outcomes](#)

- Financial models
- Motivations for cloud adoption
- Business risks
- Rationalization of the digital estate

Minimum scope

The duties of the people on the cloud operations team involve delivering maximum workload performance and minimum business interruptions within an agreed-upon operations budget.

- Determine workload criticality, impact of disruptions, or performance degradation.
- Establish business-approved cost and performance commitments.
- Monitor and operate cloud workloads.

Deliverables

- Maintain asset and workload inventory
- Monitor performance of workloads
- Maintain operational compliance
- Protect workloads and associated assets
- Recover assets if there is performance degradation or business interruption
- Mature functionality of core platforms
- Continuously improve workload performance
- Improve budgetary and design requirements of workloads to fit commitments to the business

Meeting cadence

The cloud operations team should be involved in release planning and cloud center of excellence planning to provide feedback and prepare for operational requirements.

Out of scope

Traditional IT operations that focus on maintaining current-state operations for low-level technical assets is out of scope for the cloud operations team. Things like storage, CPU, memory, network equipment, servers, and virtual machine hosts require continuous maintenance, monitoring, repair, and remediation of issues to maintain peak operations. In the cloud, many of these capital costs and operations activities are transferred to the cloud provider.

Next steps

As adoption and operations scale, it's important to define and automate governance best practices that extend existing IT requirements. Forming a cloud center of excellence is an important step to scaling cloud adoption, cloud operations, and cloud governance efforts.

Learn more about:

- [Cloud center of excellence](#) functions.
- [Organizational antipatterns: silos and fiefdoms](#).

Learn to align responsibilities across teams by developing a cross-team matrix that identifies responsible, accountable, consulted, and informed (RACI) parties. Download and modify the [RACI template ↗](#).

Cloud center of excellence (CCoE) functions

Article • 02/28/2023

Many IT organizations share the core objective of achieving business and technical agility. A cloud center of excellence (CCoE) is a function that helps organizations balance speed and stability while they pursue this objective.

Function structure

A CCoE model requires collaboration between each of the following resources:

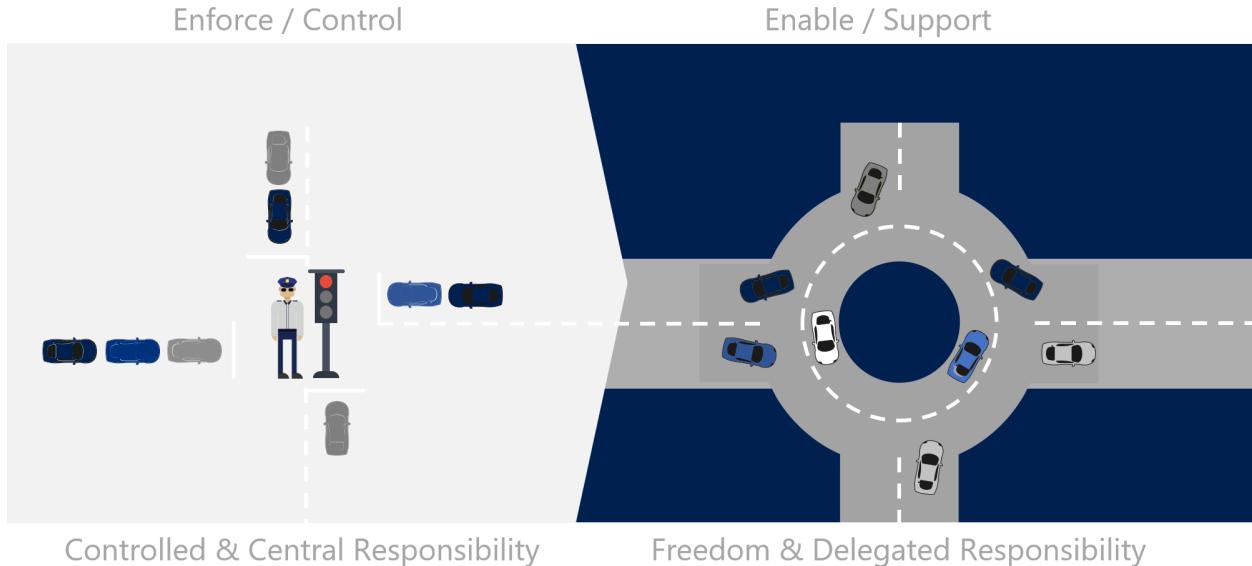
- Cloud adoption (solution architects)
- Cloud strategy (the program and project managers)
- Cloud governance
- Cloud platform
- Cloud automation

Effects

When this function is properly structured and supported, the participants can accelerate innovation and migration efforts while reducing the overall cost of change and increasing business agility. When successfully implemented, this function can produce noticeable reductions in time to market. As team practices mature, quality indicators improve, including reliability, performance efficiency, security, maintainability, and customer satisfaction. These gains in efficiency, agility, and quality are especially vital if the company plans to implement large-scale cloud migration efforts or wants to use the cloud to drive innovations that are associated with market differentiation.

When successful, a CCoE model creates a significant shift in IT. In a CCoE approach, IT serves as a broker, partner, or representative to the business. This model is a paradigm shift away from the traditional view of IT as an operations unit or abstraction layer between the business and IT assets.

The following image provides an analogy for this change. Without a CCoE approach, IT tends to focus on providing control and central responsibility, acting like the stoplights at an intersection. When the CCoE is successful, IT's role resembles a roundabout at an intersection where the focus is on freedom and delegated responsibility.



Both approaches are valid; they're alternative views of responsibility and management. A CCoE model can fit within the technology strategy if you want to establish a self-service model that allows business units to make their own decisions while adhering to a set of guidelines and established, repeatable controls.

Key responsibilities

The primary duty of the CCoE team is to accelerate cloud adoption through cloud-native or hybrid solutions.

The objective of the CCoE is to:

- Help build a modern IT organization by using agile approaches to capture and implement business requirements.
- Use reusable deployment packages that align with security, compliance, and management policies.
- Maintain a functional Azure platform in alignment with operational procedures.
- Review and approve the use of cloud-native tools.
- Standardize and automate commonly needed platform components and solutions over time.

Meeting cadence

It's important to allow for organic collaboration and to track growth through a common repository or solution catalog. Maximize natural interactions, but minimize meetings. Recurring meetings, such as release meetings that are hosted by the cloud adoption team, can provide data inputs. However, after this function matures, try to limit

dedicated meetings. Hosting a meeting after each release plan is shared can provide a minimum touch point for this team.

Solutions and controls

Each member of the CCoE needs to understand the necessary constraints, risks, and protections that led to the current set of IT controls. The CCoE turns that understanding into cloud-native (or hybrid) solutions or controls, which enable self-service business outcomes. As solutions are created, they're shared with other teams in the form of controls or automated processes that serve as guardrails for various efforts. Those guardrails help to guide team activities and to delegate responsibilities to the participants in migration or innovation efforts.

The following table describes some examples of this transition.

Scenario	Pre-CCoE solution	Post-CCoE solution
Provision a SQL server in production	Network, IT, and data platform teams provision components over the course of days or weeks.	The team that requires the server deploys a platform as a service (PaaS) instance of Azure SQL Database. Alternatively, deployment can use a preapproved template for all of the infrastructure as a service (IaaS) assets to the cloud in hours.
Provision a development environment	Network, IT, development, and DevOps teams agree on specifications and deploy an environment.	The development team defines their own specifications and deploys an environment based on allocated budget.
Update security requirements to improve data protection	Networking, IT, and security teams update networking devices and virtual machines (VMs) across several environments to add protections.	Cloud governance tools are used to update policies that can be applied immediately to all assets in all cloud environments.

Negotiations

An ongoing negotiation process is at the root of CCoE efforts. A CCoE team negotiates with existing IT functions to reduce central control. The trade-offs for the business in this negotiation are freedom, agility, and speed, and the value of the trade-off for existing IT teams is delivered as new solutions. New solutions provide the existing IT team with one or more of the following benefits:

- Ability to automate common issues

- Improvements in consistency with a reduction in day-to-day frustrations
- Opportunity to learn and deploy new technical solutions
- Reductions in high-severity incidents (requiring fewer quick fixes or late-night pager-duty responses)
- Ability to broaden their technical scope and address broader topics
- Participation in higher-level business solutions, addressing the effects of technology
- Reduction in menial maintenance tasks
- Increase in technology strategy and automation

In exchange for these benefits, the existing IT function might trade the following values:

- Sense of control from manual approval processes
- Sense of stability from change control
- Sense of job security from completion of necessary, repetitive tasks
- Sense of consistency from adherence to existing IT solution vendors

In healthy cloud-forward companies, this negotiation process is a dynamic conversation between peers and partnering IT teams. The technical details can be complex, but they're manageable when IT understands the objective and is supportive of the CCoE efforts. When IT is less than supportive, the following section on enabling CCoE success can help overcome frictions.

Enable CCoE success

Before you proceed with this model, consider the company's tolerance for a growth mindset and IT's comfort level with releasing central responsibilities. As mentioned earlier, a CCoE exchanges control for agility and speed.

This type of change takes time, experimentation, and negotiation. There will be bumps and set backs during the process, but if the team stays diligent and isn't discouraged from experimentation, there's a high probability of success in improving agility, speed, and reliability. One of the biggest success factors is support from leadership and key stakeholders.

Key stakeholders

IT leadership is the first and most obvious stakeholder. IT managers play an important part, but implementing this model requires the support of the CIO and other executive-level IT leaders.

Less obvious is the need for business stakeholders. Business agility and time to market are primary motivations for forming a CCoE. As such, the key stakeholders have a vested interest in these areas. Examples of business stakeholders include line-of-business leaders, finance executives, operations executives, and business product owners.

Support from business stakeholders

Support from the business stakeholders can accelerate CCoE efforts. Much of the focus of CCoE efforts is centered around making long-term improvements to business agility and speed. Defining the effects of current operating models and the value of improvements is valuable as a guide and negotiation tool for the CCoE. We suggest establishing or clearly defining in documentation the following items for raising support for a CCoE:

- Expected business outcomes and goals.
- Current IT process pain points, such as speed, agility, stability, and cost challenges.
- Historical effects of those pain points, such as lost market share, competitor gains in features and functions, poor customer experiences, and budget increases.
- Business improvement opportunities that are blocked by the current pain points and operating models.
- Timelines and metrics that are related to those opportunities.

These data points aren't an attack on IT. Instead, they help the CCoE team to learn from the past, establish a realistic backlog, and plan for improvement.

Ongoing support and engagement from stakeholders

CCoE teams can demonstrate quick returns in some areas, but the higher-level goals, like business agility and time to market, can take much longer. During maturation, there's a high risk of the CCoE team becoming discouraged or for members to be pulled to focus on other IT efforts.

During the first six to nine months of CCoE efforts, we recommend that business stakeholders meet monthly with IT leadership and the CCoE. There's little need for formal ceremony to these meetings. Simply reminding the CCoE members and their leadership of the importance of this program can go a long way toward CCoE success.

We also recommend that the business stakeholders stay informed of the progress and the blocking issues that the CCoE team experiences. Their efforts might seem like

technical minutiae, but business stakeholders need to understand the progress of the plan so that they can engage when the team loses steam or becomes distracted by other priorities.

Support from IT stakeholders

Support from IT stakeholders should include the following activities:

- **Support the vision:** A successful CCoE effort requires a great deal of negotiation with existing IT team members.

When done well, all of IT contributes to the solution and feels comfortable with the change. Occasionally, some members of the existing IT team might want to hold on to control mechanisms. When such situations occur, support for the CCoE by IT stakeholders is vital to the success of the CCoE. IT stakeholders need to encourage and reinforce the overall goals of the CCoE to resolve blocks to proper negotiation. On rare occasions, IT stakeholders might even need to step in and break up a deadlock or a tied vote to maintain the progress of the CCoE.

- **Maintain focus:** A CCoE can be a significant commitment for any resource-constrained IT team.

Removing strong architects from short-term projects to focus on long-term gains can create difficulty for team members who aren't part of the CCoE. IT leadership and IT stakeholders need to stay focused on the goal of the CCoE. The support of IT leaders and IT stakeholders can deprioritize the disruptions of day-to-day operations in favor of CCoE duties.

- **Create a buffer:** The CCoE team experiments with new approaches.

Some new approaches won't align well with existing operations or technical constraints. The CCoE team might experience pressure or recourse from other teams when experiments fail. It's important to encourage and buffer the CCoE team from the consequences of "fast fail" learning opportunities. It's equally important to hold the team accountable to a growth mindset to ensure that they learn from those experiments and find better solutions.

Next steps

A CCoE model requires cloud platform functions and cloud automation functions. The next step is to align cloud platform functions.

Cloud platform functions

Cloud platform functions

Article • 02/28/2023

The cloud introduces many technical changes as well as opportunities to streamline technical solutions. But general IT principles and business needs stay the same. You still need to protect sensitive business data. If your IT platform depends on a local area network, there's a good chance that you'll need network definitions in the cloud. Users who need to access applications and data will want their current identities to access relevant cloud resources.

While the cloud presents the opportunity to learn new skills, your current architects should be able to directly apply their experiences and subject matter expertise. Cloud platform functions are usually provided by a select group of architects who focus on learning about the cloud platform. These architects then aid others in decision making and the proper application of controls to cloud environments.

The skills needed to provide full platform functionality can be provided by:

- Enterprise architecture
- IT operations
- IT governance
- IT infrastructure
- Networking
- Identity
- Virtualization
- Business continuity and disaster recovery
- Application owners within IT

Preparation

- [Foundations for Cloud Architecture](#): A Pluralsight course to help architect the right foundational solutions.
- [Microsoft Azure Architecture](#): A Pluralsight course to ground architects in Azure architecture.
- [Azure network services](#): Learn Azure networking basics and how to improve resiliency and reduce latency.

Review the following:

- [Business outcomes](#)
- [Financial models](#)

- Motivations for cloud adoption
- Business risks
- Rationalization of the digital estate

Minimum scope

Cloud platform duties center around the creation and support of your cloud platform or landing zones.

The following tasks are typically executed on a regular basis:

- Monitor adoption plans and progress against the [prioritized migration backlog](#).
- Identify and prioritize platform changes that are required to support the migration backlog.

Meeting cadence:

Cloud platform expertise usually comes from a working team. Expect participants to commit a large portion of their daily schedules to cloud platform work. Contributions aren't limited to meetings and feedback cycles.

Deliverables

- Build and maintain the cloud platform to support solutions.
- Define and implement the platform architecture.
- Operate and manage the cloud platform.
- Continuously improve the platform.
- Keep up with new innovations in the cloud platform.
- Bring new cloud functionality to support business value creation.
- Suggest self-service solutions.
- Ensure solutions meet existing governance and compliance requirements.
- Create and validate deployment of platform architecture.
- Review release plans for sources of new platform requirements.

Next steps

As your cloud platform becomes better defined, aligning [cloud automation functions](#) can accelerate adoption. It can also help establish best practices while reducing business and technical risks.

Learn to align responsibilities across teams by developing a cross-team matrix that identifies responsible, accountable, consulted, and informed (RACI) parties. Download and modify the [RACI template](#).

Cloud automation functions

Article • 02/28/2023

During cloud adoption efforts, cloud automation functions unlock the potential of DevOps and a cloud-native approach. Expertise in each of these areas can accelerate adoption and innovation.

The skills needed to provide cloud automation functions can be provided by:

- DevOps engineers
- Developers with DevOps and infrastructure expertise
- IT engineers with DevOps and automation expertise

These subject matter experts might be providing functions in other areas such as cloud adoption, cloud governance, or cloud platform. After they demonstrate proficiency at automating complex workloads, you can recruit these experts to deliver automation value.

Preparation

Before you admit a team member to this group, they should demonstrate three key characteristics:

- Expertise in any cloud platform with a special emphasis on DevOps and automation.
- A growth mindset or openness to changing the way IT operates today.
- A desire to accelerate business change and remove traditional IT roadblocks.

Minimum scope

The primary duty of cloud automation is to own and advance the solution catalog. The solution catalog is a collection of prebuilt solutions or automation templates. These solutions can rapidly deploy various platforms as required to support needed workloads. These solutions are building blocks that accelerate cloud adoption and reduce the time to market during migration or innovation efforts.

Examples of solutions in the catalog include:

- A script to deploy a containerized application.
- A Resource Manager template to deploy a SQL HA AO cluster.
- Sample code to build a deployment pipeline using Azure DevOps.

- An Azure DevTest Labs instance of the corporate ERP for development purposes.
- Automated deployment of a self-service environment commonly requested by business users.

The solutions in the solution catalog aren't deployment pipelines for a workload. Instead, you might use automation scripts in the catalog to quickly create a deployment pipeline. You might also use a solution in the catalog to quickly provision platform components to support workload tasks like automated deployment, manual deployment, or migration.

Strategic tasks

- Rationalization of the digital estate:
 - Monitor adoption plans and progress against the prioritized migration backlog.
 - Identify opportunities to accelerate cloud adoption, reduce effort through automation, and improve security, stability, and consistency.
 - Prioritize a backlog of solutions for the solution catalog that delivers the most value given other strategic inputs.
- Review release plans for sources of new automation opportunities.

Meeting cadence:

Cloud automation is a working team. Expect participants to commit a large portion of their daily schedules to cloud automation work. Contributions aren't limited to meetings and feedback cycles.

The cloud automation team should align activities with other areas of capability. This alignment might result in meeting fatigue. To ensure cloud automation has sufficient time to manage the solution catalog, you should review meeting cadences to maximize collaboration and minimize disruptions to development activities.

Deliverables

- Curate or develop solutions based on the prioritized backlog.
- Ensure solutions align to platform requirements.
- Ensure solutions are consistently applied and meet existing governance and compliance requirements.
- Create and validate solutions in the catalog.

Next steps

As essential cloud functions align, the collective teams can help develop necessary technical skills.

Understand cloud data functions

Article • 02/28/2023

There are multiple audiences involved in an analytics conversation, including the typical seller, database architect, and infrastructure team. In addition, analytics solutions involve influencers, recommenders, and decision-makers from enterprise architecture, data science, business analysts, and executive leadership roles.

Azure Synapse Analytics enables the entire business, from the IT stakeholder to the business analyst, to collaborate on analytics solutions and understand cloud data functions. The following sections discuss these roles in more detail.

Database administrators and architects

Database administrators and architects are responsible for integrating and routing data sources into a centralized repository. These experts also handle the administration and performance required for the system, and the accessibility and efficiency of query and analytic modeling against that data.

Using Azure Synapse Analytics, database administrators can match their expanding responsibilities for data warehouses and data lakes. They can use familiar languages and tools, such as T-SQL, to run as many workloads as they want. They can assign resources to escalate critical workloads based on intelligent workload importance, workload isolation, and enhanced concurrency capabilities.

Infrastructure teams

These teams deal with the provisioning and architecture of the underlying compute resources required for large analytics systems. In many cases, they are managing transitions between datacenter-based and cloud-based systems, and current needs for interoperability across both. Disaster recovery, business continuity, and high availability are common concerns.

With Azure Synapse Analytics, IT professionals can protect and manage their organization's data more efficiently. They can enable big data processing with both on-demand and provisioned compute. Through tight integration with Azure Active Directory, the service helps secure access to cloud and hybrid configurations. IT professionals can enforce privacy requirements by using data masking along with row-level and column-level security.

Enterprise architects and data engineers

These teams are responsible for putting together complex solutions with components spanning integration across a wide swath of data tools and solutions. These include:

- Structured and unstructured data
- Transformation
- Storage and retrieval
- Analytic modeling
- Message-based middleware
- Data marts
- Geo-redundancy and data consistency
- Dashboards and reporting

Enterprise architects and data engineers are concerned with building effective architectures that work in an integrated manner. Such architectures preserve performance, availability, ease of administration, flexibility/extensibility, and actionability.

Data engineers can use Azure Synapse Analytics to simplify the steps to wrangle multiple data types from multiple sources, including streaming, transactional, and business data. They can use a code-free visual environment to connect to data sources and ingest, transform, and place data in the data lake.

Data scientists

Data scientists understand how to build advanced models for huge volumes of critical, yet often disparate data. Their work involves translating the needs of the business into the technology requirements for normalization and transformation of data. They create statistical and other analytical models, and ensure that line-of-business teams can get the analysis they need to run the business.

Using Azure Synapse Analytics, data scientists can build proofs of concept in minutes, and create or adjust end-to-end solutions. They can provision resources as needed, or query existing resources on demand across massive amounts of data. They can do their work in various languages, including T-SQL, R, Python, Scala, .NET, and Spark SQL.

Business analysts

These teams build and use dashboards, reports, and other forms of data visualization to gain rapid insights required for operations. Often, each line-of-business department will

have dedicated business analysts who gather and package information and analytics from specialized applications. These specialized applications can be for credit cards, retail banking, commercial banking, treasury, marketing, and other organizations.

Using Azure Synapse Analytics, business analysts can securely access datasets and use Power BI to build dashboards. They can also securely share data within and outside their organization through Azure Data Share.

Executives

Executives are responsible for charting strategy and ensuring strategic initiatives are implemented effectively across both IT and line-of-business departments. Solutions must be cost-effective, prevent disruption to the business, allow for easy extensibility as requirements change and grow, and deliver results to the business.

Mature team structures

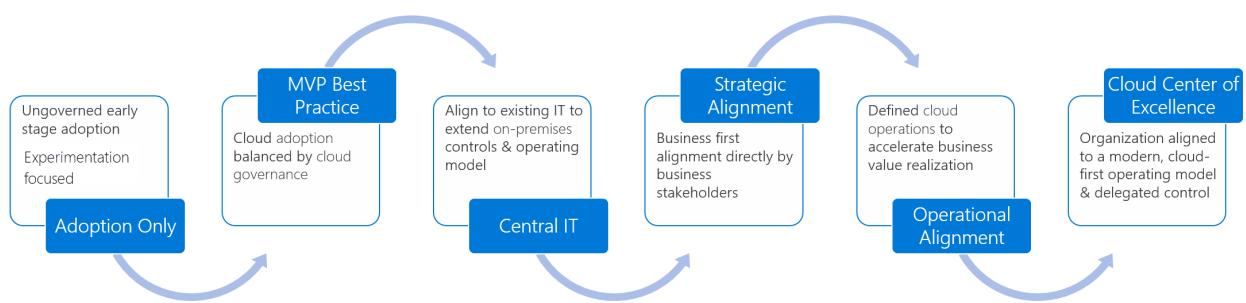
Article • 02/28/2023

All cloud functions are provided by someone during every cloud adoption effort. These assignments and team structures can develop organically, or they can be intentionally designed to match a defined team structure.

As adoption needs grow, so does the need for balance and structure. Watch this video to get an overview of common team structures at various stages of organizational maturity.

<https://www.microsoft.com/en-us/videoplayer/embed/RE4wvTS?postJs||Msg=true>

The following graphic outlines those structures based on typical maturation stages. Use these examples to find the organizational structure that best aligns with your operational needs.



Organizational structures tend to move through the common maturity model that's outlined here:

1. [Cloud adoption team only](#)
2. [MVP best practice](#)
3. [Central IT team](#)
4. [Strategic alignment](#)
5. [Operational alignment](#)
6. [Cloud center of excellence \(CCoE\)](#)

Most companies start with little more than a *cloud adoption team*. But we recommend that you establish an organizational structure that more closely resembles the [MVP best practice](#) structure.

Cloud adoption team only

The nucleus of all cloud adoption efforts is the cloud adoption team. This team drives the technical changes that enable adoption. Depending on the objectives of the

adoption effort, this team might include a diverse range of team members who handle a broad set of technical and business tasks.



For small-scale or early-stage adoption efforts, this team might be as small as one person. In larger-scale or late-stage efforts, it's common to have several cloud adoption teams, each with around six engineers. Regardless of size or tasks, the consistent aspect of any cloud adoption team is that it provides the means to onboarding solutions into the cloud. For some organizations, this might be a sufficient organizational structure. The [cloud adoption team](#) article provides more insight into the structure, composition, and function of the cloud adoption team.

⚠️ Warning

Operating with only a cloud adoption team (or multiple cloud adoption teams) is considered an antipattern and should be avoided. At a minimum, consider the [MVP best practice](#).

Best practice: minimum viable product (MVP)



We recommend that you have two teams to create balance across cloud adoption efforts. These two teams are responsible for various functions throughout the adoption effort.

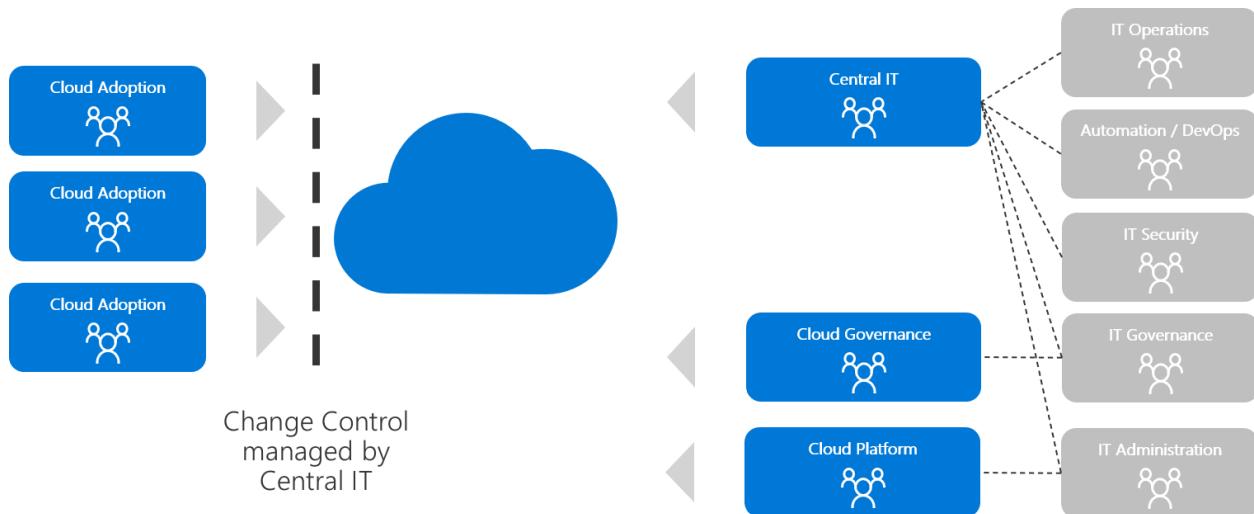
- **Cloud adoption team:** This team is accountable for technical solutions, business alignment, project management, and operations for solutions that are adopted.
- **Cloud governance team:** To balance the cloud adoption team, a cloud governance team is dedicated to ensuring excellence in the solutions that are adopted. The

cloud governance team is accountable for platform maturity, platform operations, governance, and automation.

This proven approach is considered an MVP because it might not be sustainable. Each team is wearing many hats, as outlined in the [responsible, accountable, consulted, informed \(RACI\) charts](#).

The following sections describe a fully staffed, proven organizational structure, along with approaches to aligning the appropriate structure to your organization.

Central IT team



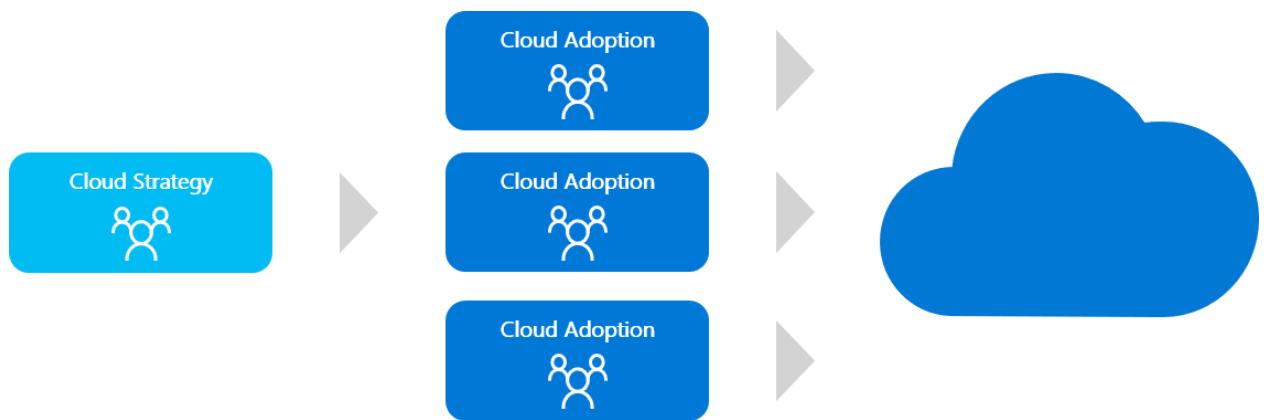
As adoption scales, the cloud governance team might struggle to keep pace with the flow of innovation from multiple cloud adoption teams. This is especially true in environments that have heavy compliance, operations, or security requirements. At this stage, it's common for companies to shift cloud responsibilities to an existing central IT team. If that team can reassess tools, processes, and people to better support cloud adoption at scale, then including the central IT team can add significant value. Subject matter experts from operations, automation, security, and administration to modernize the central IT team can drive effective operational innovations.

Unfortunately, the central IT team phase can be one of the riskiest phases of organizational maturity. The central IT team must come to the table with a strong growth mindset. If the team views the cloud as an opportunity to grow and adapt, then it can provide great value throughout the process. However, if the central IT team views cloud adoption primarily as a threat to their existing model, then the central IT team becomes an obstacle to the cloud adoption teams and the business objectives they support. Some central IT teams have spent months or even years attempting to force the cloud into alignment with on-premises approaches, with only negative results. The cloud doesn't require that everything change within the central IT team, but it does

require significant change. If resistance to change is prevalent within the central IT team, this phase of maturity can quickly become a cultural antipattern.

Cloud adoption plans heavily focused on platform as a service (PaaS), DevOps, or other solutions that require less operations support are less likely to see value during this phase of maturity. On the contrary, these types of solutions are the most likely to be hindered or blocked by attempts to centralize IT. A higher level of maturity, like a [cloud center of excellence \(CCoE\)](#), is more likely to yield positive results for those types of transformational efforts. To understand the differences between centralized IT in the cloud and a CCoE, see [Cloud center of excellence](#).

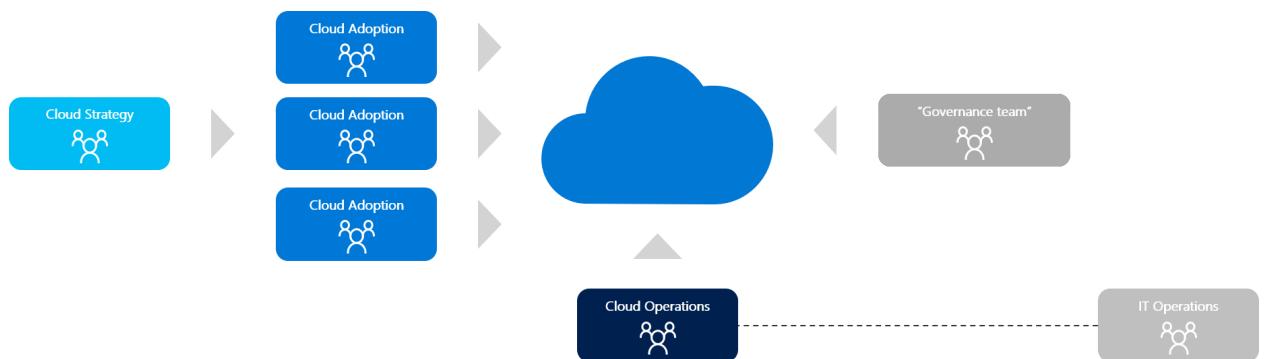
Strategic alignment



As the investment in cloud adoption grows and business values are realized, business stakeholders often become more engaged. A defined cloud strategy team aligns those business stakeholders to maximize the value realized by cloud adoption investments.

When maturity happens organically, as a result of IT-led cloud adoption efforts, strategic alignment is preceded by a governance or central IT team. When cloud adoption efforts are lead by the business, the focus on operating model and organization tends to happen earlier. Whenever possible, define business outcomes and the cloud strategy team early in the process.

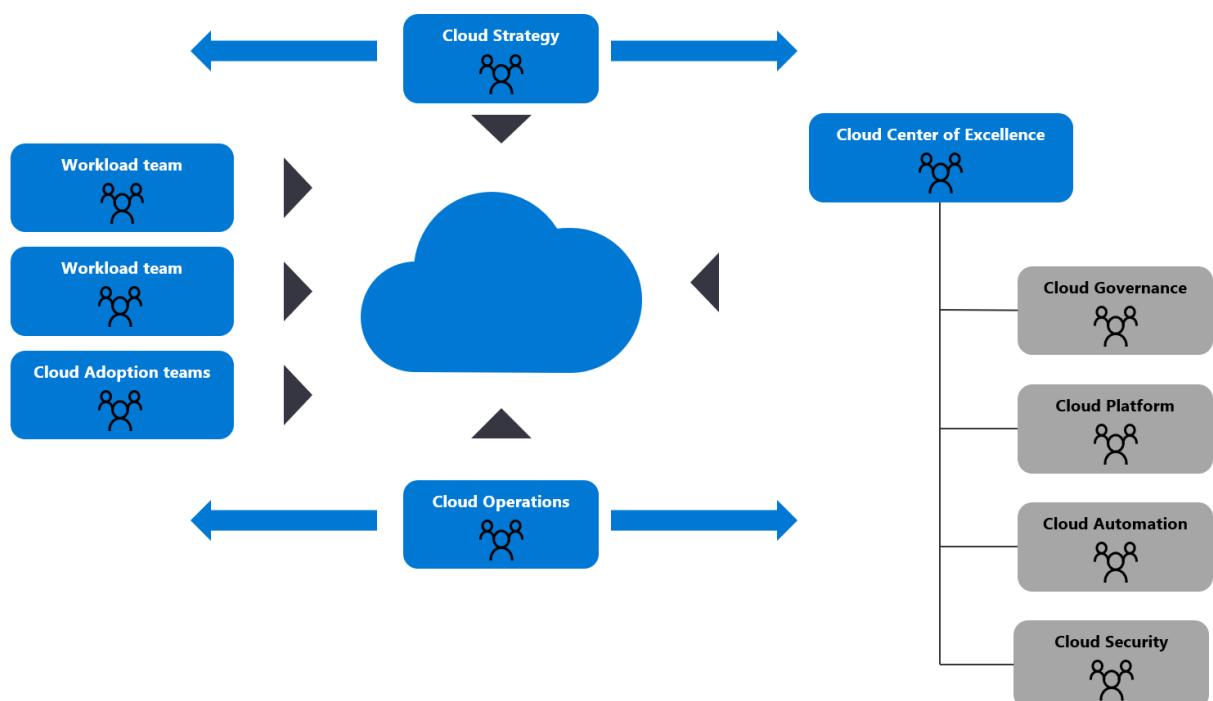
Operational alignment



Realizing business value from cloud adoption efforts requires stable operations. Operations in the cloud might require new tools, processes, or skills. When stable IT operations are required to achieve business outcomes, it's important to add a defined cloud operations team, as shown here.

Cloud operations can be delivered by the existing IT operations roles. However, it's not uncommon for cloud operations to be delegated to other parties outside of IT operations. Managed service providers, DevOps teams, and business unit IT often assume the responsibilities associated with cloud operations, with support and guardrails provided by IT operations. This is increasingly common for cloud adoption efforts that focus heavily on DevOps or PaaS deployments.

Cloud center of excellence



At the highest state of maturity, a cloud center of excellence aligns teams around a modern cloud-first operating model. This approach provides centralized IT functions like governance, security, platform, and automation.

The primary difference between this structure and the central IT team structure is a strong focus on self-service and democratization. The teams in this structure organize with the intent of delegating control as much as possible. Aligning governance and compliance practices to cloud-native solutions creates guardrails and protection mechanisms. Unlike the central IT team model, the cloud-native approach maximizes innovation and minimizes operational overhead. For this model to be adopted, mutual agreement to modernize IT processes will be required from business and IT leadership. This model is unlikely to occur organically and often requires executive support.

Next steps

After aligning to a certain stage of organizational structure maturity, you can use [RACI charts](#) to align accountability and responsibility across each team.

[Align the appropriate RACI chart](#)

Align responsibilities across teams

Article • 02/28/2023

Learn to align responsibilities across teams. Align responsibilities by developing a cross-team matrix that identifies *responsible*, *accountable*, *consulted*, and *informed* (RACI) parties. This article provides an example RACI matrix for the organizational structures described in the [Establish team structures](#) article:

- Cloud adoption team only
- MVP best practice
- Central IT team
- Strategic alignment
- Operational alignment
- Cloud center of excellence (CCoE)

To track organizational structure decisions over time, download and modify the [RACI template](#).

The examples in this article specify these RACI constructs:

- The one team that is *accountable* for a function.
- The teams that are *responsible* for the outcomes.
- The teams that should be *consulted* during planning.
- The teams that should be *informed* when work is completed.

The last row of each table (except the first) contains a link to the most-aligned cloud capability for additional information.

Cloud adoption team only

Team	Solution delivery	Business alignment	Change management	Solution operations	Governance	Platform maturity	Platform operations	Platform automation
Cloud adoption team	Accountable	Accountable	Accountable	Accountable	Accountable	Accountable	Accountable	Accountable

Best practice: Minimum viable product (MVP)

Team	Solution delivery	Business alignment	Change management	Solution operations	Governance	Platform maturity	Platform operations	Platform automation
Cloud adoption team	Accountable	Accountable	Accountable	Accountable	Consulted	Consulted	Consulted	Informed
Cloud governance team	Consulted	Informed	Informed	Informed	Accountable	Accountable	Accountable	Accountable
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	CCoE and cloud governance	CCoE and cloud platform	CCoE and cloud platform	CCoE and cloud automation

Central IT team

Team	Solution delivery	Business alignment	Change management	Solution operations	Governance	Platform maturity	Platform operations	Platform automation
Cloud adoption team	Accountable	Accountable	Responsible	Responsible	Informed	Informed	Informed	Informed
Cloud governance team	Consulted	Informed	Informed	Informed	Accountable	Consulted	Responsible	Informed
Central IT team	Consulted	Informed	Accountable	Accountable	Responsible	Accountable	Accountable	Accountable
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	Cloud governance	Central IT team	Central IT team	Central IT team

Strategic alignment

Team	Solution delivery	Business alignment	Change management	Solution operations	Governance	Platform maturity	Platform operations	Platform automation
Cloud strategy team	Consulted	Accountable	Accountable	Consulted	Consulted	Informed	Informed	Informed
Cloud adoption team	Accountable	Consulted	Responsible	Accountable	Informed	Informed	Informed	Informed
CCoE model RACI	Consulted	Informed	Informed	Informed	Accountable	Accountable	Accountable	Accountable
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	CCoE and cloud governance	CCoE and cloud platform	CCoE and cloud platform	CCoE and cloud automation

Operational alignment

Team	Solution delivery	Business alignment	Change management	Solution operations	Governance	Platform maturity	Platform operations	Platform automation
Cloud strategy team	Consulted	Accountable	Accountable	Consulted	Consulted	Informed	Informed	Informed
Cloud adoption team	Accountable	Consulted	Responsible	Consulted	Informed	Informed	Informed	Informed
Cloud operations team	Consulted	Consulted	Responsible	Accountable	Consulted	Informed	Accountable	Consulted

Team	Solution delivery	Business alignment	Change management	Solution operations	Governance	Platform maturity	Platform operations	Platform automation
CCoE model RACI	Consulted	Informed	Informed	Informed	Accountable	Accountable	Responsible	Accountable
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	CCoE and cloud governance	CCoE and cloud platform	CCoE and cloud platform	CCoE and cloud automation

Cloud center of excellence (CCoE)

Team	Solution delivery	Business alignment	Change management	Solution operations	Governance	Platform maturity	Platform operations	Platform automation
Cloud strategy team	Consulted	Accountable	Accountable	Consulted	Consulted	Informed	Informed	Informed
Cloud adoption team	Accountable	Consulted	Responsible	Consulted	Informed	Informed	Informed	Informed
Cloud operations team	Consulted	Consulted	Responsible	Accountable	Consulted	Informed	Accountable	Consulted
Cloud governance team	Consulted	Informed	Informed	Consulted	Accountable	Consulted	Responsible	Informed
Cloud platform team	Consulted	Informed	Informed	Consulted	Consulted	Accountable	Responsible	Responsible
Cloud automation team	Consulted	Informed	Informed	Informed	Consulted	Responsible	Responsible	Accountable
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	CCoE and cloud governance	CCoE and cloud platform	CCoE and cloud platform	CCoE and cloud automation

Next steps

To track decisions about organization structure over time, download and modify the RACI template. Copy and modify the most closely aligned sample from the RACI matrices in this article.

[Download the RACI template](#)

Build technical skills

Article • 12/12/2024

Organizational and environmental (technical) readiness can require new skills for technical and nontechnical contributors. The following information can help your organization build the necessary skills.

Organizational readiness learning paths

Depending on the motivations and business outcomes that are associated with a cloud-adoption effort, leaders may need to establish new organizational structures or virtual teams to facilitate various functions. The following articles can help your organization develop the necessary skills to structure those teams to meet the desired outcomes:

- [Organization alignment exercises](#): Get an overview of alignment and team structures to help meet specific goals.
- [Break down silos and fiefdoms](#): Learn about two common organizational antipatterns and ways to guide the teams to productive collaboration.

Environmental (technical) readiness learning paths

During the readiness phase, technical staff have to create a migration landing zone to host, operate, and govern workloads that they migrate to the cloud. Use the following paths to accelerate development of the necessary skills:

- [Create an Azure account](#): The first step to using Azure is to create an account. Your account holds the Azure services that you provision and handles your personal settings, like identity, billing, and preferences.
- [Azure portal](#): Tour the Azure portal features and services, and customize the portal.
- [Introduction to Azure](#): Get started with Azure. Create and configure your first virtual machine in the cloud.
- [Introduction to security in Azure](#): Learn the basic concepts to protect your infrastructure and data in the cloud. Understand what responsibilities are yours and what Azure handles.
- [Manage resources in Azure](#): Learn how to work through the Azure CLI and web portal to create, manage, and control cloud-based resources.
- [Create a VM](#): Use the Azure portal to create a virtual machine.

- [Azure network services](#): Learn Azure networking basics and how to improve resiliency and reduce latency.
- [Azure compute options](#): Review the Azure compute services.
- [Secure resources with Azure RBAC](#): Use Azure role-based access control (Azure RBAC) to secure resources.
- [Azure Storage options](#): Learn about the benefits of Azure data storage.

During the readiness phase, architects have to design solutions that span all Azure environments. The following resources can prepare them for these tasks:

- [Foundations for Cloud Architecture](#) : A Pluralsight course to help architect the right foundational solutions.
- [Microsoft Azure Architecture](#) : A Pluralsight course to ground architects in Azure architecture.
- [Designing Migrations for Microsoft Azure](#) : A Pluralsight course to help architects design a migration solution.

Deeper skills exploration

The following information describes resources for additional learning.

Typical mappings of cloud IT roles

Microsoft and partners offer various options for all audiences to develop skills with Azure services.

- [Map roles and skills](#): A resource for mapping your cloud career path. Learn about your cloud role and suggested skills. Follow a learning curriculum at your own pace to build the skills that you need most to stay relevant.
- Explore [Azure certification training and exams](#) to gain official recognition for your Azure knowledge.

Microsoft Learn

Microsoft Learn is a new approach to learning. Readiness for the new responsibilities that come with cloud adoption doesn't come easily. Microsoft Learn provides a rewarding approach to hands-on learning that helps you achieve your goals faster. Earn points, reach new levels, and achieve more.

The following are a few examples of role-specific learning paths:

- **Business users** might experience a steep learning curve when they help plan, test, and adopt cloud-based technology. Learn modules focus on adopting cloud models and tools for better managing business through cloud-based services.
- **Solution architects** can access hundreds of modules and learning paths. The available topics range from core infrastructure services to advanced data transformation.
- **Administrators** have access to modules that focus on Azure fundamentals, configuring containers, and even advanced administration in the cloud.
- **Developers** can use Microsoft Learn training resources to help during architecture, governance, modernization activities.

Learn more

For additional learning paths, browse the [Microsoft Learn training catalog](#). Use the roles filter to align learning paths with your role.

Feedback

Was this page helpful?



Build a cost-conscious organization

Article • 02/28/2023

As outlined in [Motivations: why are we moving to the cloud?](#), there are many sound reasons for a company to adopt the cloud. When cost reduction is a primary driver, it's important to create a cost-conscious organization.

Ensuring cost consciousness is not a one-time activity. Like other cloud-adoption topics, it's iterative. The following diagram outlines this process to focus on three interdependent activities: *visibility*, *accountability*, and *optimization*. These processes play out at macro and micro levels, which we describe in detail in this article.



Figure 1: Outline of the cost-conscious organization.

General cost-conscious processes

- **Visibility:** For an organization to be conscious of costs, it needs visibility into those costs. Visibility in a cost-conscious organization requires consistent reporting for the teams adopting the cloud, finance teams who manage budgets, and management teams who are responsible for the costs. This visibility is accomplished by establishing:
 - The right reporting scope.
 - Proper resource organization (management groups, resource groups, subscriptions).
 - Clear tagging strategies.
 - Proper access controls (Azure RBAC).

- **Accountability:** Accountability is as important as visibility. Accountability starts with clear budgets for adoption efforts. Budgets should be well established, clearly communicated, and based on realistic expectations. Accountability requires an iterative process and a growth mindset to drive the right level of accountability.
- **Optimization:** Optimization is the action that creates cost reductions. During optimization, resource allocations are modified to reduce the cost of supporting various workloads. This process requires iteration and experimentation. Each reduction in cost reduces performance. Finding the right balance between cost control and end-user performance expectations demands input from multiple parties.

The following sections describe the roles that various teams play in developing a cost-conscious organization.

Cloud strategy team

Building cost consciousness into cloud-adoption efforts starts at the leadership level. To be effective in the long term, the [cloud strategy team](#) should include a member of the finance team. If your financial structure holds business managers accountable for solution costs, they should be invited to join the team as well. In addition to the core activities that are typically assigned to the cloud strategy team, all members of the cloud strategy team should also be responsible for:

- **Visibility:** The cloud strategy team and [cloud governance team](#) need to know the actual costs of the cloud-adoption efforts. Given the executive-level view of this team, they should have access to multiple cost scopes to analyze spending decisions. Typically, an executive needs visibility into the total costs across all *cloud spend*. But as active members of the cloud strategy team, they should also be able to view costs per business unit or per billing unit to validate showback, chargeback, or other [cloud accounting models](#).
- **Accountability:** Budgets should be established between the cloud strategy, [cloud governance](#), and [cloud adoption](#) teams based on expected adoption activities. When deviations from budget occur, the cloud strategy team and the cloud governance team must partner to quickly determine the best course of action to remediate the deviations.
- **Optimization:** During optimization efforts, the cloud strategy team can represent the investment and return value of specific workloads. If a workload has strategic value or financial impact on the business, cost-optimization efforts should be monitored closely. If there's no strategic impact on the organization and no

inherent cost for poor performance of a workload, the cloud strategy team may approve over-optimization. To drive these decisions, the team must be able to view costs on a per-project scope.

Cloud adoption team

The [cloud adoption team](#) is at the center of all adoption activities. So, they're the first line of defense against overspending. This team has an active role in all three phases of cost-consciousness.

- **Visibility:**
 - **Awareness:** It's important for the cloud adoption team to have visibility into the cost-saving goals of the effort. Simply stating that the cloud-adoption effort will help reduce costs is a recipe for failure. *Specific* visibility is important. For example, if the goal is to reduce datacenter TCO by 3 percent or annual operating expenses by 7 percent, disclose those targets early and clearly.
 - **Telemetry:** This team needs visibility into the impact of their decisions. During migration or innovation activities, their decisions have a direct effect on costs and performance. The team needs to balance these two competing factors. Performance monitoring and cost monitoring that's scoped to the team's active projects are important to provide the necessary visibility.
- **Accountability:** The cloud adoption team needs to be aware of any preset budgets that are associated with their adoption efforts. When real costs don't align with the budget, there's an opportunity to create accountability. Accountability doesn't equate to penalizing the adoption team for exceeding budget, because budget excess can result from necessary performance decisions. Instead, accountability means educating the team about the goals and how their decisions affect those goals. Additionally, accountability includes providing a dialog in which the team can communicate about decisions that led to overspending. If those decisions are misaligned with the goals of the project, this effort provides a good opportunity to partner with the cloud strategy team to make better decisions.
- **Optimization:** This effort is a balancing act, as optimization of resources can reduce the performance of the workloads that they support. Sometimes anticipated or budgeted savings can't be realized for a workload because the workload doesn't perform adequately with the budgeted resources. In those cases, the cloud adoption team has to make wise decisions and report changes to the cloud strategy team and the cloud governance team so that budgets or optimization decisions can be corrected.

Cloud governance team

Generally, the [cloud governance team](#) is responsible for cost management across the entire cloud-adoption effort. As outlined in the [Cost Management discipline](#) topic of the Govern methodology of the Cloud Adoption Framework, cost management is the first of the Five Disciplines of Cloud Governance. Those articles outline a-series of deeper responsibilities for the cloud governance team.

This effort focuses on the following activities that are related to the development of a cost-conscious organization:

- **Visibility:** The cloud governance team works as a peer of the cloud strategy team to plan cloud-adoption budgets. These two teams also work together to regularly review actual expenses. The cloud governance team is responsible for ensuring consistent, reliable cost reporting and performance telemetry.
- **Accountability:** When budget deviations occur, the cloud strategy team and the cloud governance team must partner to quickly determine the best course of action to remediate the deviations. Generally, the cloud governance team will act on those decisions. Sometimes the action may be simple retraining for the affected [cloud adoption team](#). The cloud governance team can also help optimize deployed assets, change discounting options, or even implement automated cost-control options like blocking deployment of unplanned assets.
- **Optimization:** After assets are migrated to or created in the cloud, you can employ monitoring tools to assess performance and utilization of those assets. Proper monitoring and performance data can identify assets that should be optimized. The cloud governance team is responsible for ensuring that the monitoring and cost-reporting tools are consistently deployed. They can also help the adoption teams identify opportunities to optimize based on performance and cost telemetry.

Cloud center of excellence

While not typically responsible for cost management, the CCoE can have a significant impact on cost-conscious organizations. Many foundational IT decisions affect costs at scale. When the CCoE does their part, costs can be reduced for multiple cloud-adoption efforts.

- **Visibility:** Any management group or resource group that houses core IT assets should be visible to the CCoE team. The team can use this data to farm opportunities to optimize.

- **Accountability:** While not typically accountable for cost, the CCoE can hold itself accountable for creating repeatable solutions that minimize cost and maximize performance.
- **Optimization:** Given the CCoE's visibility to multiple deployments, the team is in an ideal position to suggest optimization tips and to help adoption teams better tune assets.

Next steps

Practicing these responsibilities at each level of the business helps drive a cost-conscious organization. To begin acting on this guidance, review the [organizational readiness introduction](#) to help identify the right team structures.

[Identify the right team structures](#)

Cloud organizational antipatterns

Article • 03/22/2023

Customers often experience cloud adoption antipatterns within their organizational structure. Many factors can cause these problems:

- Toolsets
- Partners
- Engineers
- Misaligned IT departments

It's important to understand the role of these factors in a successful cloud adoption scenario.

Antipattern: Treat IT as a cost center

Many companies treat IT departments as cost centers. This approach can lead to the perception that IT doesn't add value to the company. When employees view IT as a provider rather than an enabler, they can become discouraged. It's also hard for the company to attract the right talent. Reduced motivation and long lifecycle times result. The quality of work from IT can suffer, and [silos and fiefdoms](#) can develop.

Example: Treat IT as a cost center

A corporation manages its IT department as a cost center responsible to the chief financial officer (CFO). The managing board perceives IT as a slow service provider that's one of the company's biggest cost drivers. The managing board doesn't realize that the mobility business unit is consuming most of the assets that the IT department ordered. IT purchases a datacenter for all business units to use, but the mobility business unit gets this oversized asset. The board doesn't view IT as an enabler or a partner.

Preferred outcome: View IT as an enabler

Instead of managing your IT department as a cost center, consider one of these approaches:

- [Chargeback](#): Business units treat IT costs like operating expenses in their budgets.
- [Showback or awareness-back](#): IT functions as an agent. It reports back to the business, IT attributes any direct costs to relevant business units.

Use the cloud as a tool to increase cost and business transparency. For instance, implement a [Cost Management discipline](#) to increase cost transparency. Then you'll be more aware of the cost of different business units. You'll view the IT department as an enabler for those units.

To improve transparency, focus on visibility, accountability, and optimization when moving to the cloud. For more information, see [Build a cost-conscious organization](#).

Antipattern: Invest in new technology without involving the business

IT departments often invest significant human and financial resources in building and deploying robust platforms and toolsets. But, sometimes IT fails to consider business units and their needs during design and development phases. This omission leads to new platforms with minimal relevance for business units. Employees are then hesitant to accept the new technology. Poor or slow adoption can result. Frustration also builds within IT when business units don't use its platforms.

Example: Set up a platform without involving business units

The IT department of a data analysis firm sets up and customizes an Azure platform without involving any business units. While using the platform, business unit developers:

- Realize that they don't have the permissions that they need for deployment.
- Can only use a restricted number of services.
- Issue support tickets, which lengthen approval cycles.
- Begin to doubt the new platform.

In the end, some developers purchase an Azure subscription by themselves to avoid the hassle of IT rules and regulations. Shadow IT appears. Since the firm has little control over the shadow IT, high security risks emerge.

Preferred outcome: Involve business units in decision making

Avoid creating [IT silos](#) when deploying an enterprise-ready cloud platform. Involve developers and technical decision makers (TDMs) from business units in design and development processes. To improve platform adoption, listen to business unit input.

Refer to [Start with Cloud Adoption Framework enterprise-scale landing zones](#) for Azure best practices and design principles that increase adoption speed and are tailored toward developers. Strike the right balance between compliance and flexibility. For instance, find ways to satisfy governance and security policies while keeping development environments agile.

Antipattern: Outsource core business functions

Consulting partners and managed service providers (MSPs) can play an important role in a cloud journey. But, companies should take care that the partners' and MSPs' work doesn't provide the most value in their business. Companies that outsource responsibilities to MSPs or cloud consultants shouldn't become dependent on these providers.

Example: Outsource cloud adoption and migration

A research institute has a time-critical cloud migration project. To shorten the cloud adoption journey, it hires an MSP to build up the Azure foundation and implement the migration. Instead of learning about the cloud adoption phase and building up skills, the institute chooses to hand over all Azure responsibility to the MSP. Since the institute has no cloud or Azure knowledge, the MSP takes the lead on all decisions, making the institute dependent on the MSP.

Preferred outcome: Make critical design areas the company's responsibility

Keep outsourcing in mind as a good cost-cutting strategy. But, make decisions within your company when they involve these critical design areas:

- Governance
- Risk
- Compliance
- Identity

Keep responsibility inside the company for these and other areas that are critical to your security estate. Use external partners to speed up the adoption journey. But, to avoid becoming dependent on providers, don't outsource everything.

Antipattern: Hire technical decision makers instead of developing cloud engineers

Companies place importance on finding the right personnel. As a result, they often hire or build up TDMs during initial cloud adoption phases. Successful cloud journeys rely on TDMs. But more importantly, cloud adoptions need engineers with all-hands-on-deck mentalities and deep technical skills.

Example: Hire TDMs only

A research institute hires several TDMs to lead its cloud journey. After the initial high-level-concept phase ends, the implementation phase starts. The institute then realizes that cloud deployments behave differently than on-premises deployments. It needs extra cloud engineering effort to properly implement [infrastructure as code \(IaC\)](#) concepts and policy-driven governance.

Preferred outcome: Use cloud engineers for the implementation phase

Remember that engineers are essential for properly implementing cloud automation and landing zone concepts. Responsibilities and tasks can shift significantly when you adopt service models. By shifting responsibilities to a cloud provider, you can go into production faster. You can also use TDMs for decision making, but use capable cloud engineers for tasks that require deep engineering knowledge. Then you'll realize the advantages that the cloud provides.

Next steps

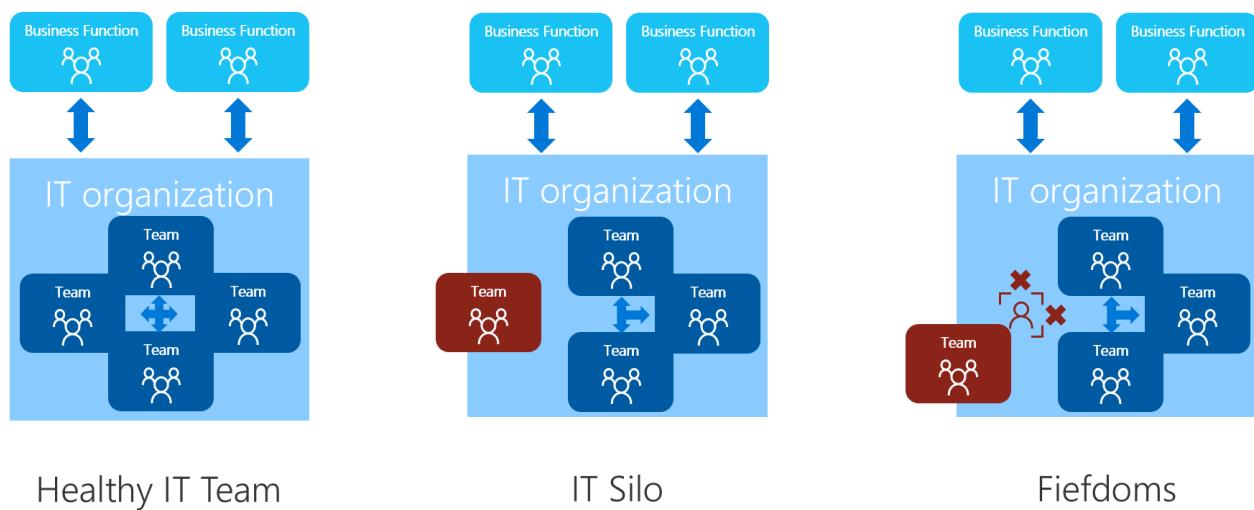
- Align responsibilities across teams
- Organizational antipatterns: Silos and fiefdoms
- Build a cost-conscious organization

Silos and fiefdoms

Article • 02/28/2023

Success in any major change to business practices, culture, or technology operations requires a growth mindset. At the heart of the growth mindset is the ability to accept change and provide leadership in spite of ambiguity.

Some antipatterns block the growth mindset in organizations that want to grow and transform. These antipatterns include micromanagement, biased thinking, and exclusionary practices. Many of these blockers are personal challenges that create personal growth opportunities for everyone. But two common antipatterns in IT, silos and fiefdoms, require more than individual growth or maturity to address.



These antipatterns are a result of organic changes within various teams, which result in unhealthy organizational behaviors. To address the resistance caused by each antipattern, it's important to understand the root cause of the formation.

Healthy, organic IT teams

It's natural to create a division of labor across IT. It's healthy to establish teams that have similar expertise, shared processes, a common objective, and an aligned vision. It's also natural for those teams to have their own microculture, shared norms, and perspectives.

Healthy IT teams focus on partnering with other teams to promote successfully completing duties. Healthy IT teams seek to understand the business goals that their technology contribution supports. The details and fiscal effects might be fuzzy, but the team's value contribution is typically understood within the team.

Although healthy IT teams have a passion for the technology that they support, they're open to change, and willing to try new things. These teams are usually the earliest and

strongest contributors to [cloud center of excellence \(CCoE\)](#) efforts. You want to heavily encourage their contribution.

Natural resistance to change

At times, the microcultures within healthy IT teams might react poorly to executive or top-down decisions to drive change. This reaction is natural, as human collectives with shared norms often cooperate to overcome external threats.

People sometimes view changes that affect the team's day-to-day jobs, sense of security, or autonomy as a risk to the collective. Signs of resistance are usually an early indicator that team members don't feel like they're part of the decision-making process.

When cloud architects and other leaders invest in abolishing personal biases and driving for inclusive IT teams, resistance to change is likely to lessen quickly and dissolve over time. A CCoE is a tool that helps cloud architects and leaders create inclusive decision making.

Healthy friction

It's easy to confuse resistance with friction. Existing IT teams are commonly knowledgeable about past mistakes, tangible risks, tribal knowledge about solutions, and undocumented technical debt. Unfortunately, even the healthiest IT teams can fall into the trap of describing these important data points as part of a specific technical solution that shouldn't be changed. This approach to communication masks the teams' knowledge and creates a perception of resistance.

Providing these teams with a mechanism for communicating in future-looking terminology adds data points, identifies gaps, and creates healthy friction around the proposed solutions. That extra friction sands down the solution's rough edges and drives longer-term values. Simply changing the conversation can create clarity around complex subjects and generate energy to deliver more successful solutions.

The guidance on [defining corporate policy](#) facilitates risk-based conversations with business stakeholders. But you can use this same model to facilitate conversations with teams that are perceived as cloud resistant. When the perception of resistance is widespread, it might be wise to include resistance resolution practices in the charter for a [cloud governance team](#).

Antipatterns

The organic and responsive growth within IT that creates healthy IT teams can also result in antipatterns that block transformation and cloud adoption. IT silos and fiefdoms are different from the natural microcultures within healthy IT teams. In either pattern, the team focus tends to be directed toward protecting their "turf". When team members are confronted with an opportunity to change and improve operations, they'll invest more time and energy into blocking the change than finding a positive solution.

As mentioned earlier, healthy IT teams can create natural resistance and positive friction. Silos and fiefdoms are a different challenge. There's no documented leading indicator for either antipattern. These antipatterns tend to be identified after months of [cloud center of excellence](#) and [cloud governance team](#) efforts. They're discovered as the result of ongoing resistance.

Even in toxic cultures, the efforts of the CCoE and the cloud governance team should help drive cultural growth and technical progress. After months of effort, a few teams might still show no signs of inclusive behaviors and stand firm in their resistance to change. These teams are likely operating in one of the following antipattern models: silos and fiefdoms. Although these models have similar symptoms, the root cause and approaches to addressing resistance is radically different between them.

IT silos

Team members in an IT silo are likely to define themselves through their alignment to a few IT vendors or an area of technical specialization. But don't confuse silos with IT fiefdoms. Silos tend to be driven by comfort and passion, and silos are sometimes easier to overcome than the fear-driven motives behind fiefdoms.

This antipattern often emerges from a common passion for a specific solution. IT silos are then reinforced by the team's advanced skills as a result of the investment in that specific solution. This superior skill is an accelerator to cloud adoption efforts if you can overcome the resistance to change. It can also become a major blocker if the silos are broken down or if the team members can't accurately evaluate options. Fortunately, you can usually overcome IT silos without making any significant changes to the organizational chart.

Address resistance from IT silos

You can address IT silos through the following approaches. The best approach depends on the root cause of the resistance.

Create virtual teams: The [Organizational readiness](#) section of the Cloud Adoption Framework describes a multilayered structure for integrating and defining four virtual

teams. One benefit of this structure is cross-organization visibility and inclusion. Introducing a [cloud center of excellence](#) creates a high-profile aspirational team that top engineers want to participate in. This change helps you create new cross-solution alignments that aren't bound by organizational-chart constraints. It also drives inclusion of top engineers who have been sheltered by IT silos.

Introducing a [cloud strategy team](#) creates immediate visibility on IT contributions regarding cloud adoption efforts. When IT silos fight for separation, this visibility can help motivate IT and business leaders to properly support those resistant team members. This process is a quick path to stakeholder engagement and support.

Consider experimentation and exposure: Team members in an IT silo have likely been constrained to think a certain way for some time. Breaking the one-track mind is a first step to addressing resistance.

Experimentation and exposure are powerful tools for breaking down barriers in silos. The team members might be resistant to competing solutions, so it's not wise to put them in charge of an experiment that competes with their existing solution. But as part of a first workload test of the cloud, the organization should implement competing solutions. The siloed team should be invited to participate as an input and review source, but not as a decision maker. Clearly communicate this approach to the team with a commitment to engage them more deeply as decision makers before moving into production solutions.

During review of the competing solution, use the practices outlined in [Define corporate policy](#) to document the experiment's tangible risks and establish policies that help the siloed team become more comfortable with the future state. This approach exposes the team to new solutions and hardens the future solution.

Be "boundary-less": The teams that drive cloud adoption find it easy to push boundaries by exploring exciting, new cloud-native solutions. It's one half of the approach to removing boundaries. But that thinking can further reinforce IT silos. Pushing for change too quickly and without respect to existing cultures can create unhealthy friction and lead to natural resistance.

When IT silos start to resist, it's important to be "boundary-less" in your own solutions. Be mindful of one simple truth: cloud-native isn't always the best solution. Consider hybrid solutions that might provide an opportunity to extend the existing investments of the IT silo into the future.

Also consider cloud-based versions of the solution that the IT silo team uses now. Experiment with the solutions and get exposure to the viewpoint of team members

working in the IT silo. At a minimum, you'll gain a fresh perspective. In many situations, you might earn enough of the IT silo's respect to lessen resistance.

Invest in education: Many people living in an IT silo became passionate about the current solution as a result of expanding their own education. Investing in the education of these teams is seldom misplaced. Allocate time for these individuals to engage in self-learning, classes, or even conferences to break the day-to-day focus on the current solution.

For education to be an investment, you must see some return from the expense. In exchange for the investment, the team might demonstrate the proposed solution to the rest of the teams involved in cloud adoption. They might also provide documentation of the tangible risks, risk management approaches, and desired policies in adopting the proposed solution. Each benefit engages the teams in the solution and uses their tribal knowledge.

Turn roadblocks into speed bumps: IT silos can slow or stop any transformation. Experimentation and iteration find a way, but only if the project keeps moving. Focus on turning roadblocks into speed bumps. Define policies that everyone can be temporarily comfortable with in exchange for continued progression.

For instance, if IT security is the roadblock because its security solution can't monitor compromised protected data in the cloud, establish data classification policies. Prevent deploying classified data into the cloud until you find an agreeable solution. Invite IT security into experimentation with hybrid or cloud-native solutions to monitor protected data.

If the network team operates as a silo, identify workloads that are self-contained and don't have network dependencies. In parallel, you can experiment, expose, and educate the network team while working on hybrid or alternative solutions.

Be patient and be inclusive: It's tempting to move on without support of an IT silo. But this decision causes disruptions and roadblocks down the road. Changing minds about the IT silo can take time. Be patient with their natural resistance. Convert it into value. Be inclusive and invite healthy friction to improve the future solution.

Never compete: The IT silo exists for a reason. It persists for a reason. There's an investment in maintaining the solution that the team members are passionate about. Directly competing with the solution or the IT silo distracts from the real goal of achieving business outcomes. This trap has blocked many transformation projects.

Stay focused on the goal, as opposed to a single component of the goal. Help accentuate the positive aspects of the IT silo's solution and help the team members

make wise decisions about the best solutions for the future. Don't insult or degrade the current solution, because that would be counterproductive.

Partner with the business: If the IT silo isn't blocking business outcomes, why do you care? There's no perfect solution or perfect IT vendor. Competition exists for a reason; each has its own benefits.

Embrace diversity and include the business by supporting and aligning to a strong [cloud strategy team](#). When an IT silo supports a solution that blocks business outcomes, it's easier to communicate that roadblock without the noise of technical squabbles.

Supporting nonblocking IT silos shows how to partner for the desired business outcomes. These efforts earn more respect and greater support from the business when an IT silo presents a legitimate blocker.

IT fiefdoms

Team members in an IT fiefdom are likely to define themselves through their alignment to a specific process or area of responsibility. The team operates under an assumption that external influence on its area of responsibility leads to problems. Fiefdoms tend to be a fear-driven antipattern, which requires significant leadership support to overcome.

Fiefdoms are especially common in organizations that have had IT downsizing, frequent turbulence in IT staff, or poor IT leadership. When the business sees IT purely as a cost center, fiefdoms are much more likely to arise.

Generally, fiefdoms are the result of a line manager who fears loss of the team and the associated power base. These leaders often have a sense of duty to their teams and feel a need to protect their subordinates from negative consequences. Phrases like "shelter the team from change" and "protect the team from process disruption" are indicators of an overly guarded manager who might need more support from leadership.

Address resistance from IT fiefdoms

IT fiefdoms can demonstrate growth by following the approaches to [addressing IT silo resistance](#). Before you try to address resistance from an IT fiefdom, we recommend that you treat the team like an IT silo first. If those types of approaches fail to yield any significant change, the resistant team might be suffering from an IT fiefdom antipattern. The root cause of IT fiefdoms is a little more complex to address, because that resistance tends to come from the direct line manager (or a leader higher up in the organization). Challenges that are IT silo-driven are typically easier to overcome.

When continued resistance from IT fiefdoms blocks cloud adoption efforts, it might be wise for a combined effort to evaluate the situation with existing IT leaders. IT leaders must carefully consider insights from the [cloud strategy team](#), [cloud center of excellence](#), and the [cloud governance team](#) before making decisions.

ⓘ Note

IT leaders should never take changes to the organizational chart lightly. They should also validate and analyze feedback from each of the supporting teams. But transformational efforts like cloud adoption tend to magnify underlying issues that have gone unnoticed or unaddressed long before this effort. When fiefdoms are preventing the company's success, leadership changes are a likely necessity.

Fortunately, removing the leader of a fiefdom doesn't always end in termination. These strong, passionate leaders can often move into a management role after a brief period of reflection. With the right support, this change is healthy for the leader of the fiefdom and the current team.

✖ Caution

For managers of IT fiefdoms, protecting the team from risk is a clear leadership value. But there's a fine line between protection and isolation. When the team is blocked from participating in driving changes, it can have psychological and professional consequences for the team. The urge to resist change might be strong, especially during times of visible change.

The manager of any isolated team can best demonstrate a growth mindset by experimenting with the guidance associated with healthy IT teams in the preceding sections. Active and optimistic participation in governance and CCoE activities can lead to personal growth. Managers of IT fiefdoms are best positioned to change stifling mindsets and help the team develop new ideas.

IT fiefdoms are sometimes a sign of systemic leadership issues. To overcome an IT fiefdom, IT leaders need the freedom to make changes to operations, responsibilities, and occasionally even the people who provide line management for specific teams. When those changes are required, it's wise to approach the changes with clear and defensible data points.

Alignment with business stakeholders, business motivations, and business outcomes might be required to drive the necessary change. Partnership with the [cloud strategy team](#), [cloud center of excellence](#), and the [cloud governance team](#) can provide the data

points needed for a defensible position. When necessary, these teams should be involved in a group escalation to address challenges that can't be addressed with IT leadership alone.

Next steps

Disrupting organizational antipatterns is a team effort. To act on this guidance, review the organizational readiness introduction to identify the right team structures and participants:

Identify the right team structures and participants

Tools and templates

Article • 05/14/2024

The Cloud Adoption Framework for Azure has tools, templates, and assessments that can help you quickly implement technical changes. Use this framework to accelerate your cloud adoption. You can use the following resources in several cloud adoption phases.

Strategy

[+] Expand table

Resource	Description
Cloud Adoption Strategy Evaluator assessment	Assess your cloud adoption strategy and get recommendations on building and advancing your cloud business case.
Cloud Journey Tracker assessment	Identify your cloud adoption path based on the needs of your business.
Strategy and plan template	Document decisions as you implement your cloud adoption strategy and plan.

Plan

[+] Expand table

Resource	Description
Strategic Migration Assessment and Readiness Tool assessment	Take the Strategic Migration Assessment and Readiness Tool (SMART) assessment to help you prepare for your Microsoft Azure migration in areas like business planning, training, security, and governance.
Cloud Journey Tracker assessment	Identify your cloud adoption path based on the needs of your business.
Strategy and plan template	Document your decisions as you implement your cloud adoption strategy and plan.
Cloud adoption plan generator	Standardize your processes. Use a template to deploy a backlog to Azure Boards.

Resource	Description
Use the Strategy-Plan-Ready-Govern Azure DevOps template ↗	Standardize your processes. Use a template to deploy a backlog to Azure Boards .

Ready

[Expand table](#)

Resource	Description
Azure naming tool ↗	Develop your comprehensive Azure naming convention in minutes.
Terraform modules	Use the Terraform open-source code base to build your Cloud Adoption Framework Azure landing zone.
Terraform registry ↗	Use Terraform to create your landing zone. Filter the Terraform registry website to list requisite Cloud Adoption Framework modules.
Enterprise-scale landing zone ↗	Deploy your open-source code base for the enterprise-scale implementation of the Cloud Adoption Framework Azure landing zone.
Data management zone ↗	Deploy a single data management zone to your subscription. Deploy the data management zone before the data landing zone.
Data landing zone ↗	Expand your landing zone with data. Data landing zone shared services include data storage, ingestion services, and management services.
Data management and landing zone Azure DevOps template ↗	Use this template to build your data management and landing zone.

Govern

[Expand table](#)

Resource	Description
Governance benchmark assessment	Identify gaps between your current state and business priorities. Find resources to help you address what's missing.

Resource	Description
Governance discipline template	Define your basic set of governance processes used to enforce each governance discipline.
Cost Management discipline template	Define your policy statements and design guidance to mature the cloud governance in your organization. This template focuses on cost management.
Deployment Acceleration discipline template	Define your policy statements and design guidance to increase the maturity of cloud governance in your organization. This template focuses on deployment acceleration.
Identity Baseline discipline template	Define your policy statements and design guidance to increase the maturity of cloud governance in your organization. This template focuses on identity requirements.
Resource Consistency discipline template	Define your policy statements and design guidance to increase the maturity of cloud governance in your organization. This template focuses on resource consistency.
Security Baseline discipline template	Define your policy statements and design guidance to increase maturity of the cloud governance in your organization. This template focuses on the security baseline.
Azure Security Benchmark	The Azure Security Benchmark (ASB) provides prescriptive best practices and recommendations to help improve the security of workloads, data, and services on Azure.
Azure Governance Visualizer	The Azure Governance Visualizer is a PowerShell script that iterates through Azure tenant's management group hierarchy down to the subscription level. It captures data from the most relevant Azure governance capabilities, such as Azure Policy and Azure role-based access control (RBAC). The visualizer shows your hierarchy map from the collected data to create a tenant summary and build granular scope insights about your management groups and subscriptions.
Azure Governance Visualizer accelerator	The Azure Governance Visualizer accelerator speeds up the adoption and deployment of the Azure Governance Visualizer script into your environment.
Microsoft Product Placemat for CMMC L3	The Microsoft Product Placemat for CMMC Level 3 (Preview) is an interactive view representing how Microsoft cloud products and services satisfy requirements for cybersecurity maturity model certification practices.
PSRule for Azure	PSRule for Azure is a set of tests and documentation to help you configure Azure solutions. You can use these tests to check your infrastructure as code (IaC) before or after deployment to Azure. PSRule for Azure includes tests that check how IaC is written and how Azure resources are configured.

Resource	Description
AzAdvertiser	Use AzAdvertiser to get Azure governance updates. For example, you can find insights about policy definitions, initiatives, aliases, security, and regulatory compliance controls in Azure Policy or Azure RBAC role definitions. You can also get insight into resource provider operations, Microsoft Entra role definitions and role actions, and Microsoft API permissions.

Migrate

[\[+\] Expand table](#)

Resource	Description
Datacenter migration discovery checklist	Use this checklist to identify workloads, servers, and other assets in your datacenter. Apply this information to help plan your migration.
Migration templates	The Azure DevOps generator includes several templates that you can use to help streamline your projects. Templates include Azure Virtual Desktop , server migration , SQL migration , and Azure Kubernetes Service (AKS) deployments .

Innovate

[\[+\] Expand table](#)

Resource	Description
Knowledge mining	The knowledge mining Azure DevOps project simplifies the process of accessing the latent insights in structured and unstructured data.
Modern data warehouse	Build your modern data warehouse by using this Azure DevOps project with links to assets, code, and learning material to help simplify your deployment.
Retail recommender	Get end-to-end guidance to enable personalized customer experiences for retail scenarios by using Azure Synapse Analytics, Azure Machine Learning services, and other Azure big data services. For more information, see Retail recommender solution accelerator .
Modern IoT Azure	Transform existing businesses and provide new businesses with growth opportunities by using connected sensors, devices, and intelligent operations. Use the Azure IoT platform to find the work items that you need to plan and implement your IoT solution.

Resource	Description
Many models solution accelerator	In the real world, many problems are too complex to solve with a single machine learning model. You might predict sales for individual stores, build a predictive maintenance model for hundreds of oil wells, or tailor the customer experience to individual users. Build a model for each instance to improve results across machine learning problems.
Demand forecasting solution accelerator	Get resources to build a solution that identifies the top factors for revenue growth from an e-commerce platform. This approach uses Azure Synapse Analytics and Machine Learning.

Manage

[Expand table](#)

Resource	Description
Microsoft Azure Well-Architected Review assessment	Define workload-specific architectures and options across your operations.
Best practices source code	Accelerate the adoption of best practices for Azure server management services. Quickly enable operations management and establish an operations baseline.
Operations management workbook	Document decisions about operations management in the cloud. Have conversations with the business to ensure alignment regarding service-level agreements (SLAs), investment in resiliency, and budget allocation related to operations.

Organize

[Expand table](#)

Resource	Description
Cross-team RACI diagram	Download and personalize the responsible, accountable, consulted, and informed (RACI) spreadsheet template to track your decisions regarding organizational structure over time.

Secure

Resource	Description
Deploy a STIG-compliant Windows virtual machine (VM)	Use a portal to deploy a Security Technical Implementation Guides (STIG)-compliant Windows VM (preview) on Azure or Azure Government.
Deploy a STIG-compliant Linux VM	Use a portal to deploy a STIG-compliant Linux VM (preview) on Azure or Azure Government.

Feedback

Was this page helpful?

 Yes

 No

Architectural decision guides

Article • 05/01/2023

The Cloud Adoption Framework's architectural decision guides describe patterns and models that can help you create your own cloud governance design. Each decision guide focuses on one core infrastructure component of cloud deployments and lists patterns and models that can support specific cloud deployment scenarios.

Actionable governance journeys provide a baseline roadmap for you to establish cloud governance for your organization. These journeys make assumptions about requirements and priorities. Your organization's situation can be different than the ones described in the governance journeys.

Architectural decision guides help you navigate these potential differences. Each guide supplements the sample governance journeys by providing alternate patterns and models, which can help you align the architectural design choices in the examples with the requirements your specific situation.

Decision guidance categories

The following architectural decision guides cover foundational technology components for all cloud deployments. Use these guides along with the cloud governance example journeys to choose a solution that suits your organization's unique needs.

[Identity](#): Integrate cloud-based identity services with your organization's identity resources to manage control and authorization within your IT environment.

[Resource consistency](#): Ensure that the deployment and organization of your cloud-based resources enforce your organization's resource management and policy requirements.

[Resource tagging](#): Organize your cloud-based resources to optimize resource utilization and cost, and to support your organization's billing models, cloud accounting approaches, and management. Resource tagging requires consistent and well-organized naming and metadata practices.

[Logging and reporting](#): Monitor log data generated by your organization's cloud-based resources. Analyzing log data provides insights into the health of operations, maintenance, and workload compliance status.

Next steps

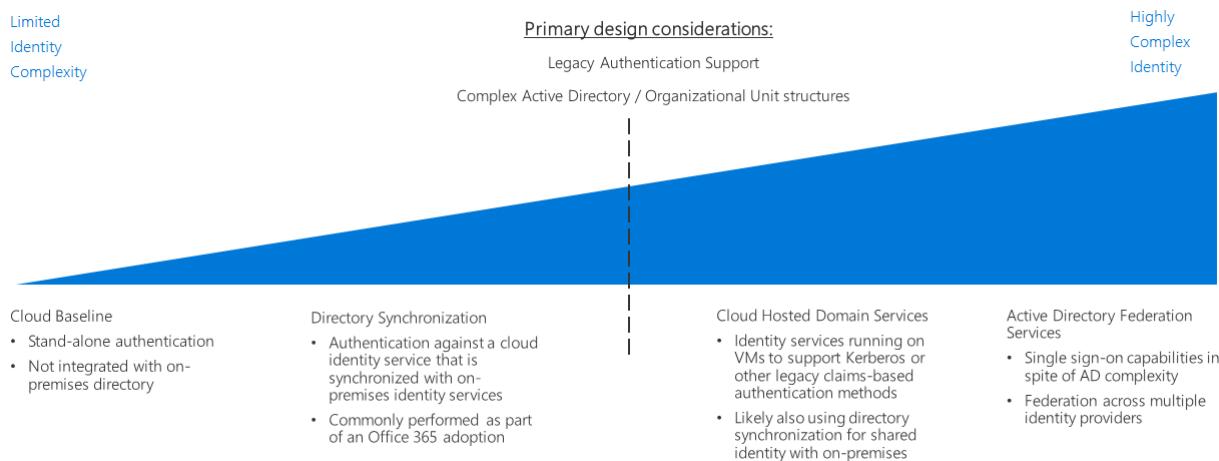
Learn about options for managing identity in your cloud environment.

[Identity decision guide](#)

Identity decision guide

Article • 10/06/2023

In any environment, whether on-premises, hybrid, or cloud-only, IT needs to control which administrators, users, and groups have access to resources. Identity and access management (IAM) services enable you to manage access control in the cloud.



Jump to: [Determine identity integration requirements](#) | [Cloud baseline](#) | [Directory synchronization](#) | [Cloud-hosted domain services](#) | [Active Directory Federation Services](#) | [Learn more](#)

Several options are available for managing identity in a cloud environment. These options vary in cost and complexity. A key factor in structuring your cloud-based identity services is the level of integration required with your existing on-premises identity infrastructure.

Microsoft Entra ID provides a base level of access control and identity management for Azure resources. If your organization's on-premises Active Directory infrastructure has a complex forest structure or customized organizational units (OUs), your cloud-based workloads might require directory synchronization with Microsoft Entra ID for a consistent set of identities, groups, and roles between your on-premises and cloud environments. Additionally, support for applications that depend on legacy authentication mechanisms might require the deployment of Active Directory Domain Services (AD DS) in the cloud.

Cloud-based identity management is an iterative process. You could start with a cloud-native solution with a small set of users and corresponding roles for an initial deployment. As your migration matures, you might need to integrate your identity solution using directory synchronization or add domains services as part of your cloud deployments. Revisit your identity strategy in every iteration of your migration process.

Determine identity integration requirements

Question	Cloud baseline	Directory synchronization	Cloud-hosted domain services	Active Directory Federation Services
Do you currently lack an on-premises directory service?	Yes	No	No	No
Do your workloads need to use a common set of users and groups between the cloud and on-premises environment?	No	Yes	No	No
Do your workloads depend on legacy authentication mechanisms, such as Kerberos or NTLM?	No	No	Yes	Yes
Do you require single sign-on across multiple identity providers?	No	No	No	Yes

As part of planning your migration to Azure, you will need to determine how best to integrate your existing identity management and cloud identity services. The following are common integration scenarios.

Cloud baseline

Microsoft Entra ID is the native identity and access management (IAM) system for granting users and groups access to management features on the Azure platform. If your organization lacks a significant on-premises identity solution, and you plan to migrate workloads to be compatible with cloud-based authentication mechanisms, you should begin developing your identity infrastructure using Microsoft Entra ID as a base.

Cloud baseline assumptions: Using a purely cloud-native identity infrastructure assumes the following:

- Your cloud-based resources will not have dependencies on on-premises directory services or Active Directory servers, or workloads can be modified to remove those dependencies.
- The application or service workloads being migrated either support authentication mechanisms compatible with Microsoft Entra ID or can be modified easily to support them. Microsoft Entra ID relies on internet-ready authentication.

mechanisms such as SAML, OAuth, and OpenID Connect. Existing workloads that depend on legacy authentication methods using protocols such as Kerberos or NTLM might need to be refactored before migrating to the cloud using the cloud baseline pattern.

💡 Tip

Completely migrating your identity services to Microsoft Entra ID eliminates the need to maintain your own identity infrastructure, significantly simplifying your IT management.

But Microsoft Entra ID is not a full replacement for a traditional on-premises Active Directory infrastructure. Directory features such as legacy authentication methods, computer management, or group policy might not be available without deploying additional tools or services to the cloud.

For scenarios where you need to integrate your on-premises identities or domain services with your cloud deployments, see the directory synchronization and cloud-hosted domain services patterns discussed below.

Directory synchronization

For organizations with existing on-premises Active Directory infrastructure, directory synchronization is often the best solution for preserving existing user and access management while providing the required IAM capabilities for managing cloud resources. This process continuously replicates directory information between Microsoft Entra ID and on-premises directory services, allowing common credentials for users and a consistent identity, role, and permission system across your entire organization.

⚠ Note

Organizations that have adopted Microsoft 365 might have already implemented **directory synchronization** between their on-premises Active Directory infrastructure and Microsoft Entra ID.

Directory synchronization assumptions: Using a synchronized identity solution assumes the following:

- You need to maintain a common set of user accounts and groups across your cloud and on-premises IT infrastructure.
- Your on-premises identity services support replication with Microsoft Entra ID.

💡 Tip

Any cloud-based workloads that depend on legacy authentication mechanisms provided by on-premises Active Directory servers and that are not supported by Microsoft Entra ID will still require either connectivity to on-premises domain services or virtual servers in the cloud environment providing these services. Using on-premises identity services also introduces dependencies on connectivity between the cloud and on-premises networks.

Cloud-hosted domain services

If you have workloads that depend on claims-based authentication using legacy protocols such as Kerberos or NTLM, and those workloads cannot be refactored to accept modern authentication protocols such as SAML or OAuth and OpenID Connect, you might need to migrate some of your domain services to the cloud as part of your cloud deployment.

This pattern involves deploying virtual machines running Active Directory to your cloud-based virtual networks to provide Active Directory Domain Services (AD DS) for resources in the cloud. Any existing applications and services migrating to your cloud network should be able to use these cloud-hosted directory servers with minor modifications.

It's likely that your existing directories and domain services will continue to be used in your on-premises environment. In this scenario, you should also use directory synchronization to provide a common set of users and roles in both the cloud and on-premises environments.

Cloud-hosted domain services assumptions: Performing a directory migration assumes the following:

- Your workloads depend on claims-based authentication using protocols like Kerberos or NTLM.
- Your workload virtual machines need to be domain-joined for management or application of Active Directory group policy purposes.

💡 Tip

While a directory migration coupled with cloud-hosted domain services provides great flexibility when migrating existing workloads, hosting virtual machines within your cloud virtual network to provide these services does increase the complexity

of your IT management tasks. As your cloud migration experience matures, examine the long-term maintenance requirements of hosting these servers. Consider whether refactoring existing workloads for compatibility with cloud identity providers such as Microsoft Entra ID can reduce the need for these cloud-hosted servers.

Active Directory Federation Services

Identity federation establishes trust relationships across multiple identity management systems to allow common authentication and authorization capabilities. You can then support single sign-on capabilities across multiple domains within your organization or identity systems managed by your customers or business partners.

Microsoft Entra ID supports federation of on-premises Active Directory domains using [Active Directory Federation Services \(AD FS\)](#). For more information about how this can be implemented in Azure, see [Extend AD FS to Azure](#).

Learn more

For more information about identity services in Azure, see:

- [Microsoft Entra ID](#). Microsoft Entra ID provides cloud-based identity services. It allows you to manage access to your Azure resources and control identity management, device registration, user provisioning, application access control, and data protection.
- [Microsoft Entra Connect](#). The Microsoft Entra Connect tool allows you to connect Microsoft Entra instances with your existing identity management solutions, allowing synchronization of your existing directory in the cloud.
- [Azure role-based access control \(Azure RBAC\)](#). Azure RBAC efficiently and securely manages access to resources in the management plane. Jobs and responsibilities are organized into roles, and users are assigned to these roles. Azure RBAC allows you to control who has access to a resource along with which actions a user can perform on that resource.
- [Microsoft Entra Privileged Identity Management \(PIM\)](#). PIM lowers the exposure time of resource access privileges and increases your visibility into their use through reports and alerts. It limits users to just-in-time privileges, assigning their privileges for a limited duration then revoking those privileges automatically.
- [Integrate on-premises Active Directory domains with Microsoft Entra ID](#). This reference architecture provides an example of directory synchronization between on-premises Active Directory domains and Microsoft Entra ID.

- [Extend Active Directory Domain Services \(AD DS\) to Azure](#). This reference architecture provides an example of deploying AD DS servers to extend domain services to cloud-based resources.
- [Extend Active Directory Federation Services \(AD FS\) to Azure](#). This reference architecture configures Active Directory Federation Services (AD FS) to perform federated authentication and authorization with your Microsoft Entra directory.

Next steps

Identity is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. To learn about alternative patterns or models used when making design decisions for other types of infrastructure, see the architectural decision guides overview.

[Architectural decision guides overview](#)

Azure role-based access control

Article • 12/01/2022

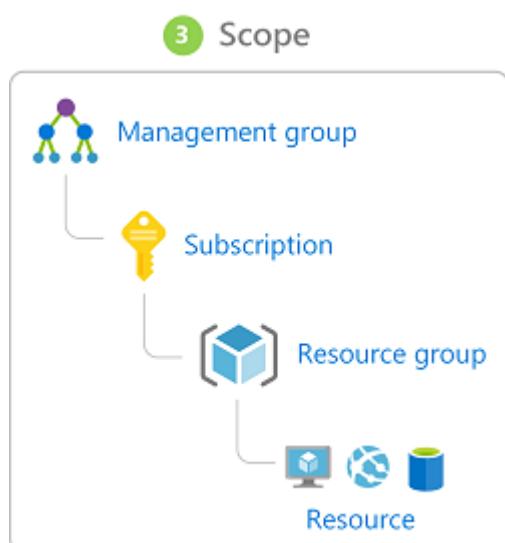
Group-based access rights and privileges are a good practice. Dealing with groups rather than individual users simplifies maintenance of access policies, provides consistent access management across teams, and reduces configuration errors. Assigning users to and removing users from appropriate groups helps keep current the privileges of a specific user. [Azure role-based access control \(Azure RBAC\)](#) offers fine-grained access management for resources organized around user roles.

For an overview of recommended Azure RBAC practices as part of an identity and security strategy, see [Azure identity management and access control security best practices](#).

Overview of Azure role-based access control

By using [Azure role-based access control](#), you can separate duties within your team and grant only enough access for specific Azure Active Directory (Azure AD) users, groups, service principals, or managed identities to perform their jobs. Instead of giving everybody unrestricted access to your Azure subscription or resources, you can limit permissions for each set of resources.

[Azure role definitions](#) list operations that are permitted or disallowed for users or groups assigned to that role. A role's [scope](#) specifies which resources these defined permissions apply to. Scopes can be specified at multiple levels: management group, subscription, resource group, or resource. Scopes are structured in a parent/child relationship.



For detailed instructions for assigning users and groups to specific roles and assigning roles to scopes, see [Add or remove Azure role assignments using the Azure portal](#).

When planning your access control strategy, use a least-privilege access model that grants users only the permissions required to perform their work. The following diagram shows a suggested pattern for using Azure RBAC through this approach.

	Role	Reader	Resource-specific or custom role	Contributor	Owner
Scope					
Subscription	 Subscription	Observers		Users managing resources	Admins
Resource group	 Resource group				
Resource	 Resource		Automated processes		

Note

The more specific or detailed permissions are that you define, the more likely it is that your access controls will become complex and difficult to manage. This is especially true as your cloud estate grows in size. Avoid resource-specific permissions. Instead, use **management groups** for enterprise-wide access control and **resource groups** for access control within subscriptions. Also avoid user-specific permissions. Instead, assign access to **groups in Azure AD**.

Use Azure built-in roles

Azure provides a many built-in role definitions, with three core roles for providing access:

- The **Owner role** can manage everything, including access to resources.
- The **Contributor role** can manage everything except access to resources.
- The **Reader role** can view everything but not make any changes.

Beginning from these core access levels, additional built-in roles provide more detailed controls for accessing specific resource types or Azure features. For example, you can manage access to virtual machines by using the following built-in roles:

- The [Virtual Machine Administrator Login role](#) can view virtual machines in the portal and sign in as `administrator`.
- The [Virtual Machine Contributor role](#) can manage virtual machines, but it can't access them or the virtual network or storage account they're connected to.
- The [Virtual Machine User Login role](#) can view virtual machines in the portal and sign in as a regular user.

For another example of using built-in roles to manage access to particular features, see the discussion on controlling access to cost-tracking features in [Track costs across business units, environments, or projects](#).

For a complete list of available built-in roles, see [Azure built-in roles](#).

Use custom roles

Although the roles built in to Azure support a wide variety of access control scenarios, they might not meet all the needs of your organization or team. For example, if you have a single group of users responsible for managing virtual machines and Azure SQL Database resources, you might want to create a custom role to optimize management of the required access controls.

The Azure RBAC documentation contains instructions on [creating custom roles](#), along with details on [how role definitions work](#).

Separation of responsibilities and roles for large organizations

Azure RBAC allows organizations to assign different teams to various management tasks within large cloud estates. It can allow central IT teams to control core access and security features, while also giving software developers and other teams large amounts of control over specific workloads or groups of resources.

Most cloud environments can also benefit from an access-control strategy that uses multiple roles and emphasizes a separation of responsibilities between these roles. This approach requires that any significant change to resources or infrastructure involves multiple roles to complete, ensuring that more than one person must review and approve a change. This separation of responsibilities limits the ability of a single person to access sensitive data or introduce vulnerabilities without the knowledge of other team members.

The following table illustrates a common pattern for dividing IT responsibilities into separate custom roles:

Group	Common role name	Responsibilities
Security operations	SecOps	<p>Provides general security oversight.</p> <p>Establishes and enforces security policy such as encryption at rest.</p> <p>Manages encryption keys.</p> <p>Manages firewall rules.</p>
Network operations	NetOps	Manages network configuration and operations within virtual networks, such as routes and peerings.
Systems operations	SysOps	Specifies compute and storage infrastructure options, and maintains resources that have been deployed.
Development, test, and operations	DevOps	<p>Builds and deploys workload features and applications.</p> <p>Operates features and applications to meet service-level agreements and other quality standards.</p>

The breakdown of actions and permissions in these standard roles are often the same across your applications, subscriptions, or entire cloud estate, even if these roles are performed by different people at different levels. Accordingly, you can create a common set of Azure role definitions to apply across different scopes within your environment. Users and groups can then be assigned a common role, but only for the scope of resources, resource groups, subscriptions, or management groups that they're responsible for managing.

For example, in a [hub and spoke network topology](#) with multiple subscriptions, you might have a common set of role definitions for the hub and all workload spokes. A hub subscription's NetOps role can be assigned to members of the organization's central IT team, who are responsible for maintaining networking for shared services used by all workloads. A workload spoke subscription's NetOps role can then be assigned to members of that specific workload team, allowing them to configure networking within that subscription to best support their workload requirements. The same role definition is used for both, but scope-based assignments ensure that users have only the access that they need to perform their job.

Hub-and-spoke network topology

Article • 12/01/2022

Hub and spoke is a networking model for efficiently managing common communication or security requirements. It also helps avoid Azure subscription limitations. This model addresses the following concerns:

- **Saving on costs and efficient management:** Centralize services that can be shared by multiple workloads, like network virtual appliances (NVAs) and DNS servers. With a single location for services, IT can minimize redundant resources and management effort.
- **Overcoming subscription limits:** Large cloud-based workloads might require using more resources than a single Azure subscription contains. Peering workload virtual networks from different subscriptions to a central hub can overcome these limits. For more information, see [Azure subscription limits](#).
- **Instituting a separation of concerns:** You can deploy individual workloads between central IT teams and workload teams.

Smaller cloud estates might not benefit from the added structure and capabilities that this model offers. But larger cloud adoption efforts should consider implementing a hub-and-spoke networking architecture if they have any of the concerns listed previously.

Note

The Azure reference architectures site contains example templates that you can use as the basis for implementing your own hub-and-spoke networks:

- [Implement a hub-and-spoke network topology in Azure](#)
- [Implement a hub-and-spoke network topology with shared services in Azure](#)

Overview

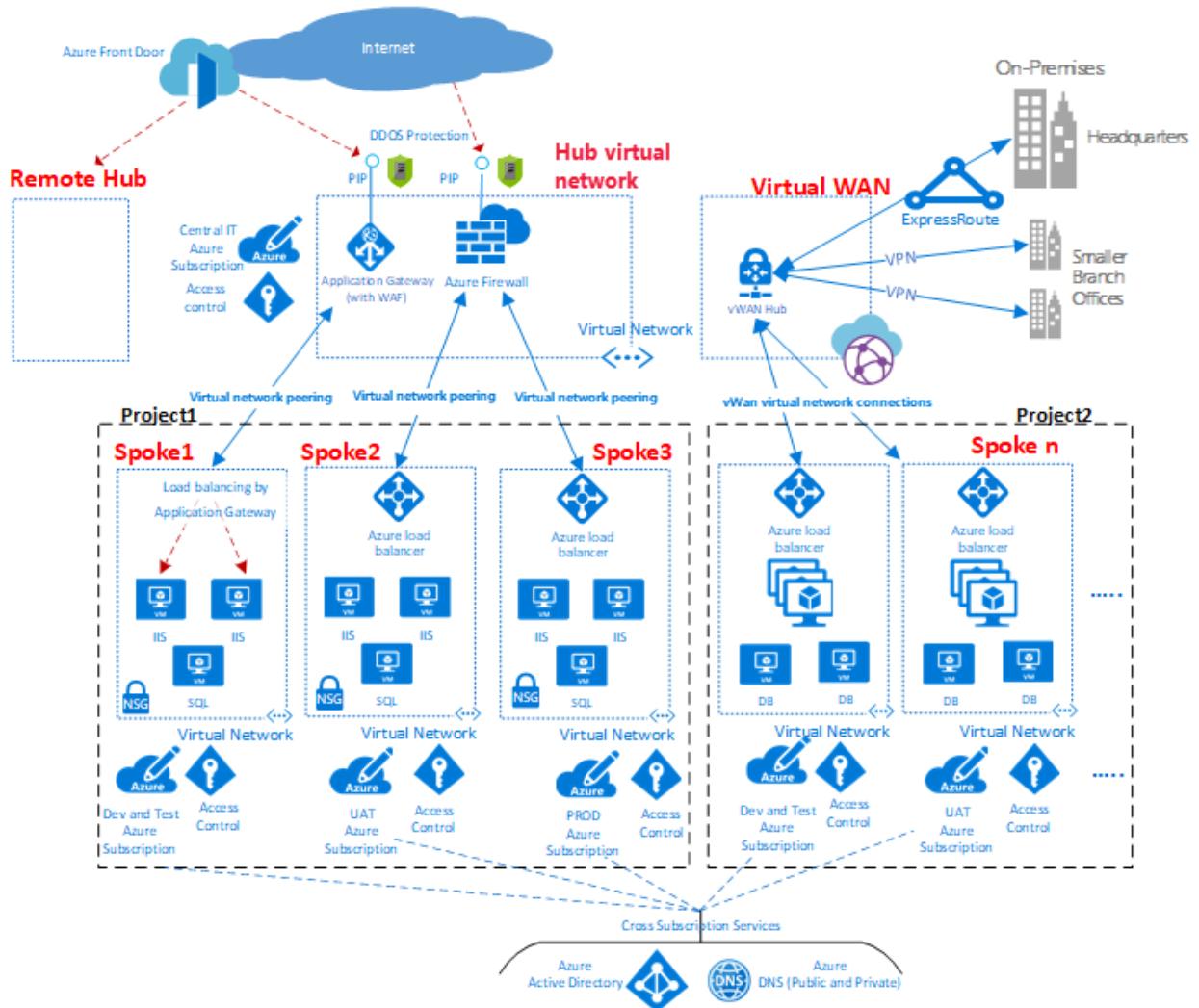


Figure 1: An example of a hub-and-spoke network topology.

As shown in the diagram, Azure supports two types of hub-and-spoke design. The first type supports communication, shared resources, and centralized security policy. This type is labeled as *VNet hub* in the diagram. The second type is based on Azure Virtual WAN, which is labeled as *Virtual WAN* in the diagram. This type is for large-scale branch-to-branch and branch-to-Azure communications.

A hub is a central network zone that controls and inspects ingress or egress traffic between zones: internet, on-premises, and spokes. The hub-and-spoke topology gives your IT department an effective way to enforce security policies in a central location. It also reduces the potential for misconfiguration and exposure.

The hub often contains the common service components that the spokes consume. Examples of common central services are:

- The Windows Server Active Directory infrastructure is required to authenticate third-party users who access untrusted networks before they access workloads in the spoke. It includes the related Active Directory Federation Services (AD FS).
- A DNS service resolves naming the workload in the spokes to access resources on-premises and on the internet if **Azure DNS** isn't used.

- A public key infrastructure implements single sign-on for workloads.
- TCP and UDP traffic flow is controlled between the spoke network zones and the internet.
- Flow is controlled between the spokes and on-premises.
- Flow is controlled between one spoke and another, if needed.

You can minimize redundancy, simplify management, and reduce overall cost by using the shared hub infrastructure to support multiple spokes.

The role of each spoke can be to host different types of workloads. The spokes also provide a modular approach for repeatable deployments of the same workloads. Examples include dev/test, user acceptance testing, staging, and production.

The spokes can also segregate and enable different groups within your organization. An example is Azure DevOps groups. Inside a spoke, it's possible to deploy a basic workload or complex multitier workloads with traffic control between the tiers.

The Application Gateway shown in the diagram above can live in spoke with the application it's serving for better management and scale. However, corporate policy might dictate you place the Application Gateway in the hub for centralized management and segregation of duty.

Subscription limits and multiple hubs

In Azure, every type of component is deployed in an Azure subscription. The isolation of Azure components in different Azure subscriptions can satisfy the requirements of different lines of business, such as setting up differentiated levels of access and authorization.

A single hub-and-spoke implementation can scale up to a large number of spokes, but as with every IT system, there are platform limits. The hub deployment is bound to a specific Azure subscription, which has restrictions and limits. One example is a maximum number of virtual network peerings. For more information, see [Azure subscription and service limits](#).

When limits might be an issue, you can scale up the architecture by extending the model to a cluster of hubs and spokes. You can connect multiple hubs in one or more Azure regions by using:

- Virtual network peering
- Azure ExpressRoute
- Azure Virtual WAN
- Site-to-site VPN

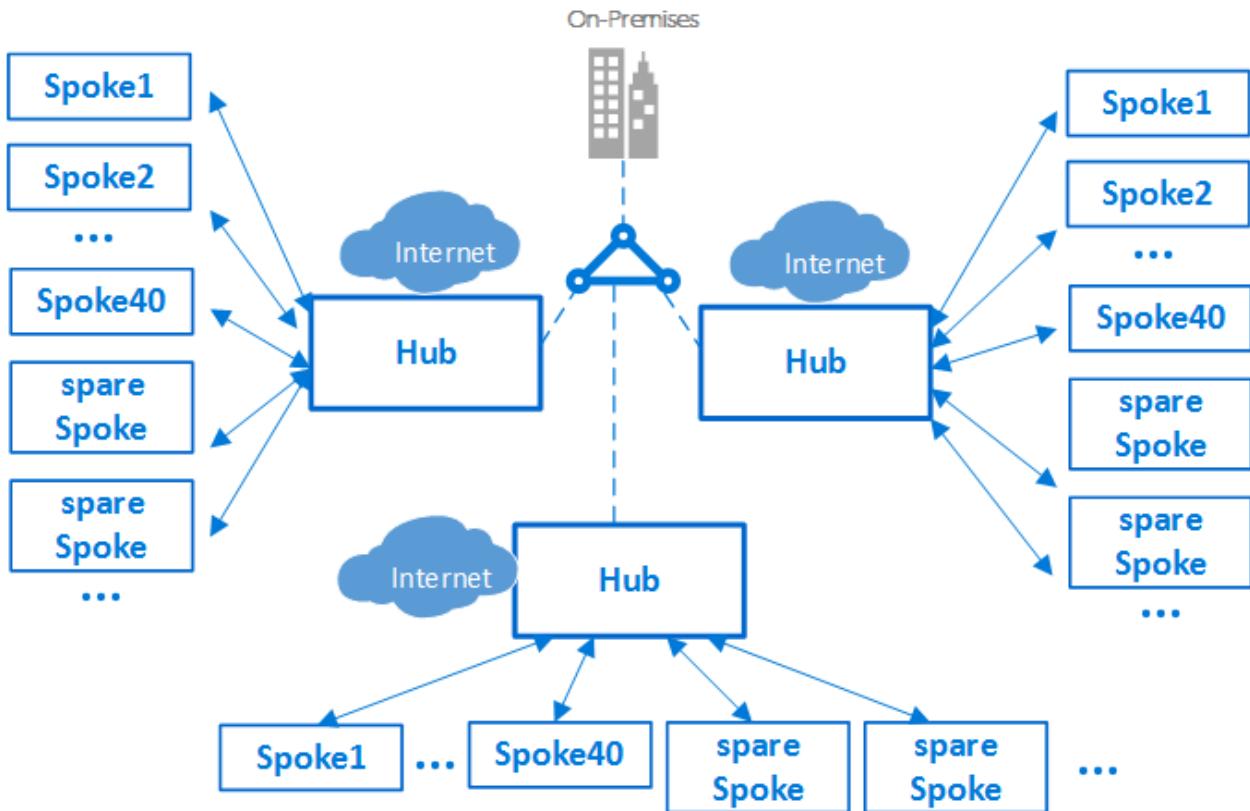


Figure 2: A cluster of hubs and spokes.

The introduction of multiple hubs increases the cost and management overhead of the system. This increase is only justified by:

- Scalability
- System limits
- Redundancy and regional replication for user performance or disaster recovery

In scenarios that require multiple hubs, all of the hubs should strive to offer the same set of services for operational ease.

Interconnection between spokes

It's possible to implement complex multitier workloads in a single spoke. You can implement multitier configurations by using subnets (one for every tier) in the same virtual network and by using network security groups to filter the flows.

An architect might want to deploy a multitier workload across several virtual networks. With virtual network peering, spokes can connect to other spokes in the same hub or in different hubs.

A typical example of this scenario is the case where application processing servers are in one spoke or virtual network. The database deploys in a different spoke or virtual network. In this case, it's easy to interconnect the spokes with virtual network peering

and avoid transiting through the hub. Do a careful architecture and security review to ensure that bypassing the hub doesn't bypass important security or auditing points that are only in the hub.

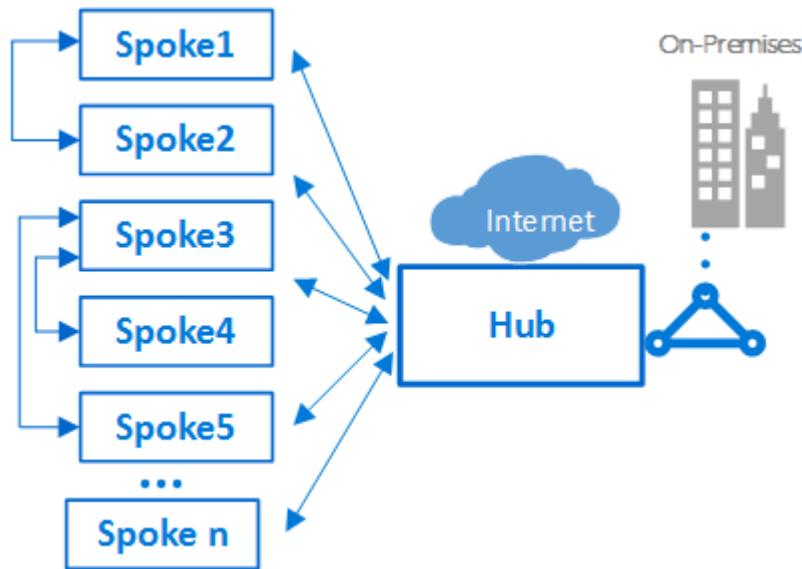


Figure 3: An example of spokes connecting to each other and a hub.

Spokes can also be interconnected to a spoke that acts as a hub. This approach creates a two-level hierarchy: the spoke in the higher level, level 0, becomes the hub of lower spokes, or level 1, of the hierarchy. The spokes are required to forward the traffic to the central hub. This requirement is so that the traffic can transit to its destination in either the on-premises network or the public internet. An architecture with two levels of hubs introduces complex routing that removes the benefits of a simple hub-and-spoke relationship.

ⓘ Note

You can use **Azure Virtual Network Manager (AVNM)** to create new or onboard existing hub and spoke virtual network topologies for central management of connectivity and security controls.

A connectivity configuration enables you to create a mesh or a hub-and-spoke network topology including direct connectivity between spoke virtual networks.

A security configuration allows you to define a collection of rules that you can apply to one or more network groups at the global level.

Next steps

Now that you've explored the best practices for networking, learn how to approach identity and access controls.

Identity Management and access control security best practices

Perimeter networks

Article • 05/02/2023

Perimeter networks, sometimes called demilitarized zones (DMZs), help provide secure connectivity between cloud networks, on-premises or physical datacenter networks, and the internet.

In effective perimeter networks, incoming packets flow through security appliances that are hosted in secure subnets, before the packets can reach back-end servers. Security appliances include firewalls, network virtual appliances (NVAs), and other intrusion detection and prevention systems. Internet-bound packets from workloads must also flow through security appliances in the perimeter network before they can leave the network.

Usually, central IT teams and security teams are responsible for defining operational requirements for perimeter networks. Perimeter networks can provide policy enforcement, inspection, and auditing.

Perimeter networks can use the following Azure features and services:

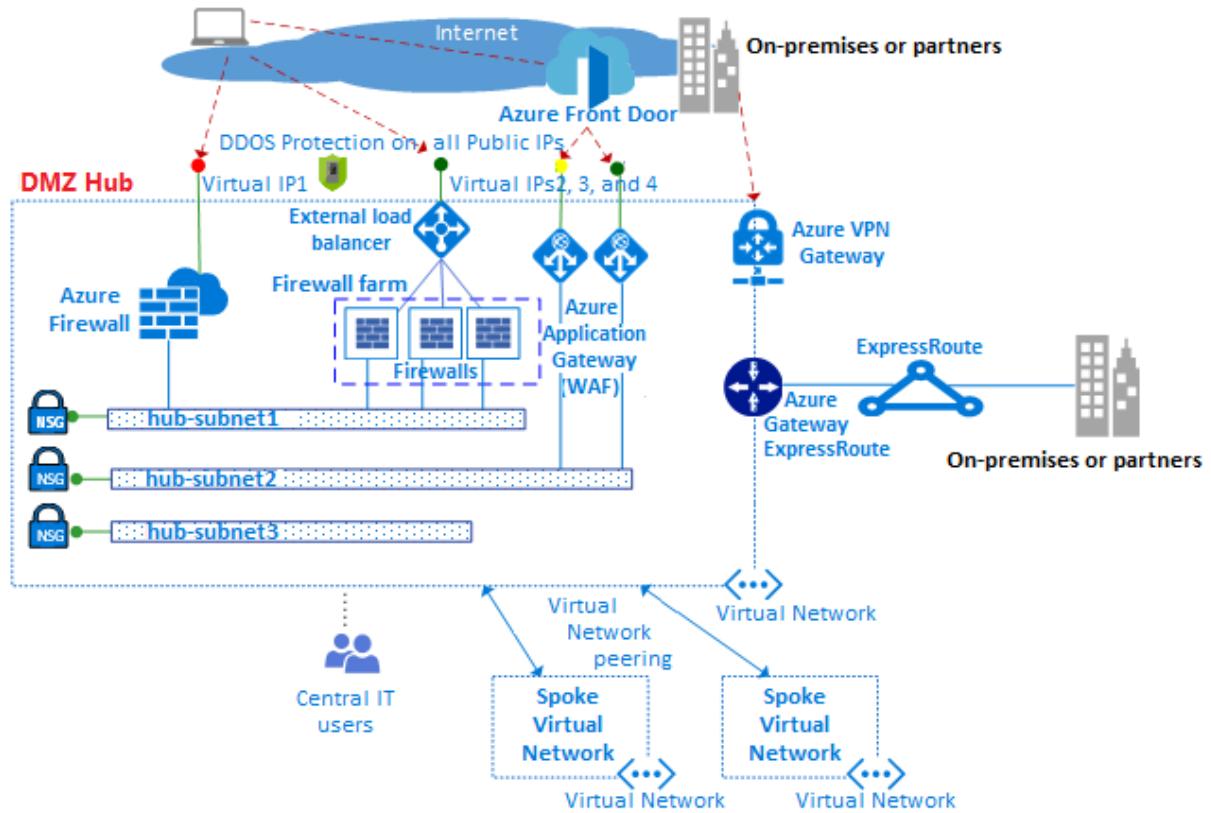
- [Virtual networks](#), [user-defined routes](#), and [network security groups \(NSGs\)](#)
- [Azure Firewall](#)
- [Azure Web Application Firewall](#) on [Azure Application Gateway](#)
- [Azure Web Application Firewall](#) on [Azure Front Door](#)
- [Other network virtual appliances \(NVAs\)](#)
- [Azure Load Balancer](#)
- [Public IP addresses](#)

For more information about perimeter networks, see [The virtual datacenter: A network perspective](#).

For example templates you can use to implement your own perimeter networks, see the reference architecture [Implement a secure hybrid network](#).

Perimeter network topology

The following diagram shows an example [hub and spoke network](#) with perimeter networks that enforce access to the internet and to an on-premises network.



The perimeter networks are connected to the DMZ hub. In the DMZ hub, the perimeter network to the internet can scale up to support many lines of business. This support uses multiple farms of web application firewalls (WAFs) and Azure Firewall instances that help protect the spoke virtual networks. The hub also allows connectivity to on-premises or partner networks via virtual private network (VPN) or Azure ExpressRoute as needed.

Virtual networks

Perimeter networks are typically built within virtual networks. The virtual network uses multiple subnets to host the different types of services that filter and inspect traffic to or from other networks or the internet. These services include NVAs, WAFs, and Application Gateway instances.

User-defined routes

In a hub and spoke network topology, you must guarantee that traffic generated by virtual machines (VMs) in the spokes passes through the correct virtual appliances in the hub. This traffic routing requires [user-defined routes](#) in the subnets of the spokes.

User-defined routes can guarantee that traffic passes through specified custom VMs, NVAs, and load balancers. The route sets the front-end IP address of the internal load balancer as the next hop. The internal load balancer distributes the internal traffic to the virtual appliances in the load balancer back-end pool.

You can use user-defined routes to direct traffic through firewalls, intrusion detection systems, and other virtual appliances. Customers can route network traffic through these security appliances for security boundary policy enforcement, auditing, and inspection.

Azure Firewall

[Azure Firewall](#) is a managed cloud-based firewall service that helps protect your resources in virtual networks. Azure Firewall is a fully stateful managed firewall with built-in high availability and unrestricted cloud scalability. You can use Azure Firewall to centrally create, enforce, and log application and network connectivity policies across subscriptions and virtual networks.

Azure Firewall uses a static public IP address for virtual network resources. External firewalls can use the static public IP to identify traffic that originates from your virtual network. Azure Firewall works with Azure Monitor for logging and analytics.

Network virtual appliances

You can manage perimeter networks with access to the internet through Azure Firewall or through a farm of firewalls or WAFs. An Azure Firewall instance and an NVA firewall can use a common administration plane with a set of security rules. These rules help protect the workloads hosted in the spokes, and control access to on-premises networks. Azure Firewall has built-in scalability, and NVA firewalls can be manually scaled behind a load balancer.

Different lines of business use many different web applications, which can suffer from various vulnerabilities and potential exploits. A WAF detects attacks against HTTP/S web applications in more depth than a generic firewall. Compared with traditional firewall technology, WAFs have a set of specific features to help protect internal web servers from threats.

A firewall farm has less specialized software than a WAF, but also has a broader application scope to filter and inspect any type of egress and ingress traffic. If you use an NVA approach, you can find and deploy software from the Azure Marketplace.

Use one set of Azure Firewall instances or NVAs for traffic that originates on the internet, and another set for traffic that originates on-premises. Using only one set of firewalls for both kinds of traffic is a security risk, because there's no security perimeter between the two sets of network traffic. Using separate firewall layers reduces the complexity of checking security rules, and clarifies which rules correspond to which incoming network requests.

Azure Load Balancer

[Azure Load Balancer](#) offers a high-availability Layer 4 transmission control protocol/user datagram protocol (TCP/UDP) load balancing service. This service can distribute incoming traffic among service instances defined by a load-balancing set. Load Balancer can redistribute traffic from front-end public or private IP endpoints, with or without address translation, to a pool of back-end IP addresses like NVAs or VMs.

Load Balancer can also probe the health of the server instances. When an instance fails to respond to a probe, the load balancer stops sending traffic to the unhealthy instance.

In the hub and spoke network topology example, you deploy an external load balancer to both the hub and the spokes. In the hub, the load balancer efficiently routes traffic to services in the spokes. The load balancer in the spokes manages application traffic.

Azure Front Door

[Azure Front Door](#) is a highly available and scalable web application acceleration platform and global HTTPS load balancer. You can use Azure Front Door to build, operate, and scale out your dynamic web application and static content. Azure Front Door runs in more than 100 locations at the edge of Microsoft's global network.

Azure Front Door provides your application with:

- Unified regional and stamp maintenance automation.
- Business continuity and disaster recovery (BCDR) automation.
- Unified client and user information.
- Caching.
- Service insights.

Azure Front Door offers performance, reliability, and service-level agreements (SLAs) for support. Azure Front Door also offers compliance certifications and auditable security practices that Azure develops, operates, and natively supports.

Application Gateway

[Application Gateway](#) is a dedicated virtual appliance that gives you a managed application delivery controller. Application Gateway offers various Layer 7 load-balancing capabilities for your application.

Application Gateway helps you optimize web farm productivity by offloading CPU-intensive secure socket layer (SSL) termination. Application Gateway also provides other

Layer 7 routing capabilities, such as:

- Round-robin distribution of incoming traffic.
- Cookie-based session affinity.
- URL path-based routing.
- Hosting multiple websites behind a single application gateway.

The [Application Gateway with WAF SKU](#) includes a WAF, and provides protection to web applications from common web vulnerabilities and exploits. You can configure Application Gateway as an internet-facing gateway, an internal-only gateway, or a combination of both.

Public IPs

With some Azure features, you can associate service endpoints to a [public IP](#) address. This option provides access to your resource from the internet. The endpoint uses network address translation (NAT) to route traffic to the internal address and port on the Azure virtual network. This path is the main way for external traffic to pass into the virtual network. You can configure public IP addresses to control what traffic passes in, and how and where it's translated into the virtual network.

Azure DDoS Protection

[Azure DDoS Protection](#) provides extra mitigation capabilities to help protect your Azure resources in virtual networks from distributed denial of service (DDoS) attacks. DDoS Protection has two SKUs, DDoS IP Protection and DDoS Network Protection. For more information, see [About Azure DDoS Protection SKU Comparison](#).

DDoS Protection is easy to enable and requires no application changes. You can tune protection policies through dedicated traffic monitoring and machine-learning algorithms. DDoS Protection applies protection to IPv4 Azure public IP addresses that are associated with resources deployed in virtual networks. Example resources include Load Balancer, Application Gateway, and Azure Service Fabric instances.

Real-time telemetry is available through Azure Monitor views, both during an attack and for historical purposes. You can add application-layer protection by using Web Application Firewall in Application Gateway.

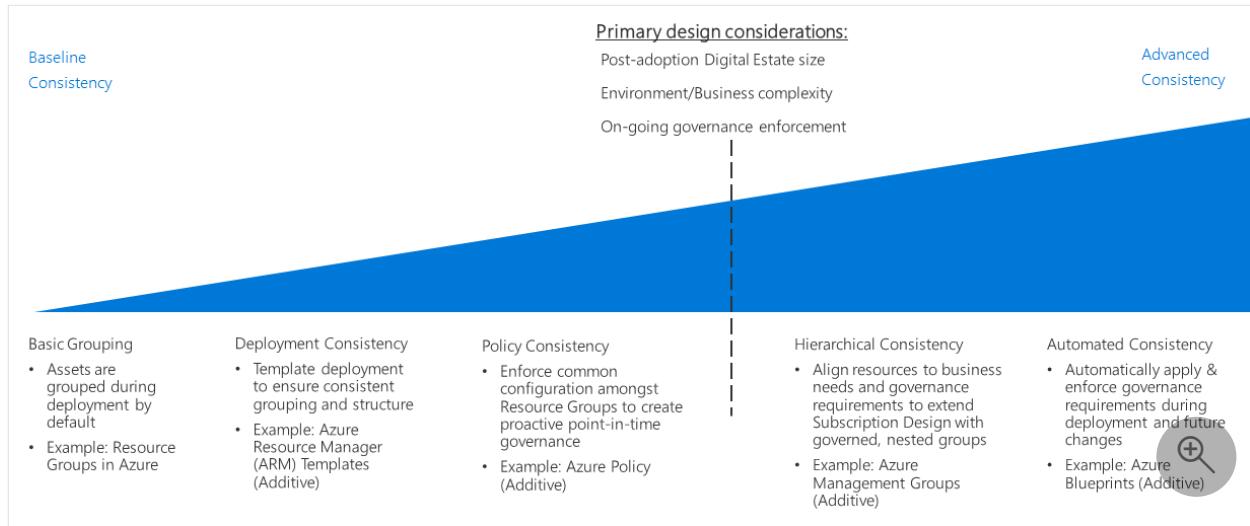
Next steps

Learn how to efficiently manage common communication or security requirements by using the [hub and spoke network topology](#) model.

Resource consistency decision guide

Article • 05/14/2024

Azure [subscription design](#) defines how you organize your cloud assets in relation to your organization's structure, accounting practices, and workload requirements. In addition to this level of structure, addressing your organizational governance policy requirements across your cloud estate requires the ability to consistently organize, deploy, and manage resources within a subscription.



Jump to: [Basic grouping](#) | [Deployment consistency](#) | [Policy consistency](#) | [Hierarchical consistency](#) | [Automated consistency](#)

Decisions regarding the level of your cloud estate's resource consistency requirements are primarily driven by these factors: post-migration digital estate size, business or environmental requirements that don't fit neatly within your existing subscription design approaches, or the need to enforce governance over time after resources have been deployed.

As these factors gain importance, the benefits of ensuring consistent deployment, grouping, and management of cloud-based resources becomes even more vital. Gaining more advanced levels of resource consistency to meet increasing requirements takes greater effort across automation, tooling, and consistency enforcement. That effort results in more time spent on change management and tracking.

Basic grouping: Resource groups

In Azure, [resource groups](#) are a core resource organization mechanism to logically group resources within a subscription.

You can use resource groups as containers for resources that have a common lifecycle and shared management constraints, such as policy or Azure role-based access control (RBAC) requirements. You can't nest resource groups, and resources can only belong to one resource group. All control plane actions affect all resources in a resource group. For example, deleting a resource group also deletes all resources within that group.

When you design or update your regional resource organization, consider the following factors. Is there a logical group of resources:

- That you can develop together?
- That you can manage, update, and monitor together? Can the same people or team carry out those tasks?
- That one team uses within a single geography/region?
- That you can retire together?

If the answer is yes for any of these questions, consider placing those resources (deployed in region X) together in a resource group (also deployed in region X).

To minimize the effect of regional outages, place resources in the same region as the resource group. For more information, see [Resource group location alignment](#).

Note

If you have resources that are in the same resource group, but the resources are in different regions, consider moving your resources to a [new resource group or subscription](#).

To [determine if your resource supports moving to another resource group](#), inventory your resources by cross-referencing them. Ensure that you meet the appropriate [prerequisites](#).

Tip

[Audit](#)  your resource group alignment with Azure Policy. [Assign a built-in Azure Policy definition](#) at the [intermediate root management group](#) level to verify whether the locations of the resources in your tenant hierarchy match the location of their respective resource groups.

Deployment consistency

When you build on top of the base resource grouping mechanism, the Azure platform provides a system for using templates to deploy your resources to the cloud environment. You can use templates to create consistent organization and naming conventions when deploying workloads. Templates enforce those aspects of your resource deployment and management design.

[Azure Resource Manager templates](#) let you repeatedly deploy your resources in a consistent state using a predetermined configuration and resource group structure. Resource Manager templates help you define a set of standards as a basis for your deployments.

For example, you can use a standard template to deploy a web server workload that contains two virtual machines as web servers combined with a load balancer to distribute traffic between the servers. You can then reuse this template to create a structurally identical set of virtual machines. The VMs have a load balancer whenever this type of workload is needed, and only changing the deployment name and IP addresses involved.

You can also programmatically deploy these templates and integrate them with your continuous integration and continuous delivery (CI/CD) systems.

Policy consistency

Part of resource grouping design involves using a common configuration when deploying resources. Using a common configuration ensures that governance policies apply when you create resources.

By combining resource groups and standardized Resource Manager templates, you can enforce standards for what settings are required in a deployment and what [Azure Policy](#) rules apply to each resource group or resource.

For example, you might have a requirement that all virtual machines deployed within your subscription connect to a common subnet managed by your central IT team. Use a standard template for deploying workload VMs to create a separate resource group for the workload and deploy the required VMs there. This resource group has a policy rule to only allow network interfaces within the resource group to be joined to the shared subnet.

For a more in-depth discussion of enforcing your policy decisions within a cloud deployment, see [Policy enforcement](#).

Hierarchical consistency

Resource groups let you support extra levels of hierarchy inside your organization within the subscription. Hierarchies support Azure Policy rules and access controls at a resource group level. As the size of your cloud estate grows, you might need to support more complicated cross-subscription governance requirements. Use the Azure Enterprise Agreement's enterprise, department, account, subscription hierarchy.

[Azure management groups](#) lets you organize subscriptions into more sophisticated organizational structures. You can group subscriptions in a hierarchy distinct from your Enterprise Agreement's hierarchy. This alternate hierarchy lets you apply access control and policy enforcement mechanisms across multiple subscriptions and the resources they contain. You can use management group hierarchies to match your cloud estate's subscriptions with operations or business governance requirements. For more information, see the [subscription decision guide](#).

Automated consistency

For large cloud deployments, global governance becomes both more important and more complex. It's crucial to automatically apply and enforce governance requirements when deploying resources, and meet updated requirements for existing deployments.

An Azure landing zone is an environment that follows key design principles across eight design areas. These design principles accommodate all application portfolios and enable application migration, modernization, and innovation at scale. For more information about Azure landing zones, see [What is an Azure landing zone?](#).

IT and development teams can use Azure landing zones to rapidly deploy new workloads and networking assets that comply with changing organizational policy requirements. Platform teams can use [infrastructure as code \(IaC\) templates](#), including [policy as code](#) practices, to deploy and manage the Azure landing zone. Incorporate these practices into your CI/CD pipelines to ensure that you apply new governance standards as you update templates and definitions.

Next step

Resource consistency is just one of the core infrastructure components that requires architectural decisions during a cloud adoption process. Visit the architectural decision guides overview to learn about the patterns and models for design decisions on various types of infrastructure.

[Architectural decision guides](#)

Feedback

Was this page helpful?

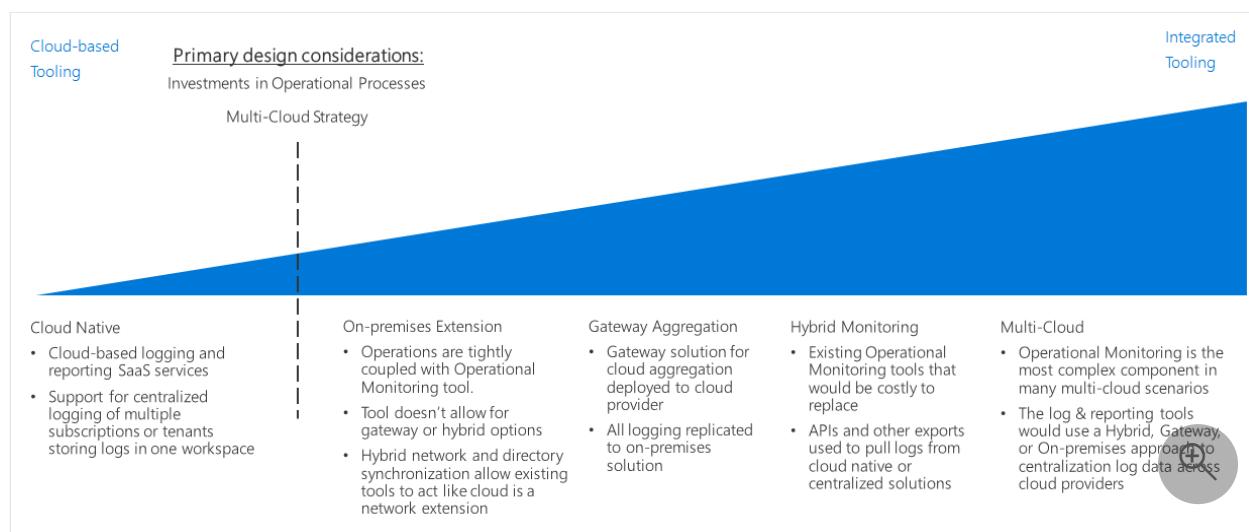
 Yes

 No

Logging and reporting decision guide

Article • 03/15/2023

All organizations need mechanisms for notifying IT teams of performance, uptime, and security issues before they become serious problems. A successful monitoring strategy shows you how the individual components that make up your workloads and networking infrastructure are performing. Within the context of a public cloud migration, integrating logging and reporting with any of your existing monitoring systems, is critical to success. Also, surfacing important events and metrics to the appropriate IT staff is vital in ensuring your organization meets its uptime, security, and policy compliance goals.



Jump to: [Planning your monitoring infrastructure](#) | [Cloud-native](#) | [On-premises extension](#) | [Gateway aggregation](#) | [Hybrid monitoring \(on-premises\)](#) | [Hybrid monitoring \(cloud-based\)](#) | [Multicloud](#) | [Learn more](#)

The inflection point when determining a cloud logging and reporting strategy is based primarily on:

- Existing investments your organization has made in operational processes.
- Any requirements you have to support a multicloud strategy.

Activities in the cloud are logged and reported in multiple ways. Cloud-native and centralized logging are two common managed service options that are driven by the subscription design and the number of subscriptions.

Plan your monitoring infrastructure

When planning your deployment, consider where your logging data stores and how you'll integrate cloud-based reporting and monitoring services with your existing

processes and tools.

Question	Cloud-native	On-premises extension	Hybrid monitoring	Gateway aggregation
Do you have an existing on-premises monitoring infrastructure?	No	Yes	Yes	No
Do you have requirements preventing storage of log data on external storage locations?	No	Yes	No	No
Do you need to integrate cloud monitoring with on-premises systems?	No	No	Yes	No
Do you need to process or filter telemetry data before submitting it to your monitoring systems?	No	No	No	Yes

Cloud-native

A cloud-native SaaS solution such as [Azure Monitor](#), is the easier choice if:

- Your organization currently lacks established logging and reporting systems.
- Your planned deployment doesn't need to be integrated with existing on-premises or other external monitoring systems.

In this scenario, all your log data records and stores in the cloud. Azure platform and Azure Monitor provide the logging and reporting tools that process and surface information to your IT staff.

As needed, implement custom logging solutions based on Azure Monitor for each subscription or workload in smaller or experimental deployments. These solutions are organized centrally to monitor log data across your entire cloud estate.

Cloud-native assumptions: Using a cloud-native logging and reporting system assumes the following circumstances:

- You don't need to integrate the log data from your cloud workloads into existing on-premises systems.
- You won't be using your cloud-based reporting systems to monitor on-premises systems.

On-premises extension

It might require substantial redevelopment effort for applications and services migrating to the cloud to use cloud-based logging and reporting solutions such as Azure Monitor. In this case, consider allowing the workloads to continue sending telemetry data to existing on-premises systems.

To support this approach, your cloud resources must communicate directly with your on-premises systems through a combination of [hybrid networking](#) and [cloud-hosted domain services](#). With this communication in place, the cloud virtual network functions as a network extension of the on-premises environment. So, cloud-hosted workloads can communicate directly with your on-premises logging and reporting system.

This approach capitalizes on your existing investment in monitoring tooling with limited modification to any cloud-deployed applications or services. Also, this approach is often the fastest way to support monitoring during a lift and shift migration. But it won't capture log data produced by cloud-based PaaS and SaaS resources. It also omits any VM-related logs generated by the cloud platform itself, such as VM status. As a result, this pattern should be a temporary solution until you implement a more comprehensive hybrid monitoring solution.

On-premises-only assumptions: Using an on-premises logging and reporting system assumes the following circumstances:

- You maintain log data only in your on-premises environment. Maintain your log data in this way either in support of technical requirements or due to regulatory or policy requirements.
- Your on-premises systems don't support hybrid logging and reporting or gateway aggregation solutions.
- Your cloud-based applications can submit telemetry directly to your on-premises logging systems. Or your monitoring agents that submit to on-premises can be deployed to workload VMs.
- Your workloads don't depend on PaaS or SaaS services that require cloud-based logging and reporting.

Gateway aggregation

A log data [gateway aggregation](#) service might be required for scenarios where:

- The amount of cloud-based telemetry data is large.
- Existing on-premises monitoring systems need log data modified before processing.

A gateway service deploys to your cloud provider. Then, you configure relevant applications and services to submit telemetry data to the gateway instead of a default

logging system. The gateway then processes the data: aggregating, combining, or otherwise formatting it before then submitting the data to your monitoring service for ingestion and analysis.

Also, you can use a gateway to aggregate and pre-process telemetry data bound for cloud-native or hybrid systems.

Gateway aggregation assumptions:

- You expect large volumes of telemetry data from your cloud-based applications or services.
- You need to format or otherwise optimize telemetry data before submitting it to your monitoring systems.
- Your monitoring systems have APIs or other mechanisms available to ingest log data after processing by the gateway.

Hybrid monitoring (on-premises)

A hybrid monitoring solution combines log data from both your on-premises and cloud resources to provide an integrated view into your IT estate's operational status.

If you have an existing investment in on-premises monitoring systems that would be difficult or costly to replace, you might need to integrate the telemetry from your cloud workloads into preexisting on-premises monitoring solutions. In a hybrid on-premises monitoring system, on-premises telemetry data continues to use the existing on-premises monitoring system. Cloud-based telemetry data is either sent to the on-premises monitoring system directly, or the data is sent to Azure Monitor then compiled and ingested into the on-premises system at regular intervals.

On-premises hybrid monitoring assumptions: Using an on-premises logging and reporting system for hybrid monitoring assumes the following conditions:

- You must use existing on-premises reporting systems to monitor cloud workloads.
- You maintain ownership of log data on-premises.
- Your on-premises management systems have APIs or other mechanisms available to ingest log data from cloud-based systems.

Tip

As part of the iterative nature of cloud migration, transitioning from distinct cloud-native and on-premises monitoring to a partial hybrid approach is likely as the integration of cloud-based resources and services into your overall IT estate matures.

Hybrid monitoring (cloud-based)

If you don't have a compelling need to maintain an on-premises monitoring system, or you want to replace on-premises monitoring systems with a centralized cloud-based solution, you can also integrate on-premises log data with Azure Monitor to provide a centralized cloud-based monitoring system.

Mirroring the on-premises centered approach, in this scenario, cloud-based workloads would submit telemetry direct to Azure Monitor. And on-premises applications and services would either submit telemetry directly to Azure Monitor, or aggregate that data on-premises for ingestion into Azure Monitor at regular intervals. Azure Monitor would then serve as your primary monitoring and reporting system for your entire IT estate.

Cloud-based hybrid monitoring assumptions: Using cloud-based logging and reporting systems for hybrid monitoring assumes the following conditions:

- You don't depend on existing on-premises monitoring systems.
- Your workloads don't have regulatory or policy requirements to store log data on-premises.
- Your cloud-based monitoring systems have APIs or other mechanisms available to ingest log data from on-premises applications and services.

Multicloud

Integrating logging and reporting capabilities across a multicloud platform can be complicated. Services offered between platforms are often not directly comparable, and logging and telemetry capabilities provided by these services differ as well.

Multicloud logging support often requires the use of gateway services to process log data into a common format before submitting data to a hybrid logging solution.

Learn more

[Azure Monitor](#) is the default reporting and monitoring service for Azure. It provides:

- A unified platform for collecting application telemetry, host telemetry (such as VMs), container metrics, Azure platform metrics, and event logs.
- Visualization, queries, alerts, and analytical tools. It can provide insights into virtual machines, guest operating systems, virtual networks, and workload application events.

- REST APIs for integration with external services and automating the monitoring and alerting services.
- Integration with many popular third-party vendors.

Next steps

Logging and reporting is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. Visit the architectural decision guides overview to learn about alternative patterns or models to use when making design decisions for other types of infrastructure.

[Architectural decision guides](#)

Organize and set up Azure Machine Learning environments

Article • 08/30/2022

When you're planning an Azure Machine Learning deployment for an enterprise environment, there are some common decision points that affect how you create the workspace:

- **Team structure:** The way you organize your data science teams and collaborate on projects, given use case and data segregation, or cost management requirements
- **Environments:** The environments you use as part of your development and release workflow to segregate development from production
- **Region:** The location of your data and the audience to which you need to serve your machine learning solution

Team structure and workspace setup

The workspace is the top-level resource in Azure Machine Learning. It stores the artifacts that are produced when working with machine learning and the managed compute and pointers to attached and associated resources. From a manageability standpoint, the workspace as an Azure Resource Manager resource supports Azure role-based access control (Azure RBAC), management by Policy, and you can use it as a unit for cost reporting.

Organizations typically choose one or a combination of the following solution patterns to follow manageability requirements.

Workspace per team: Use one workspace for each team when all members of a team require the same level of access to data and experimentation assets. For example, an organization with three machine learning teams might create three workspaces, one for each team.

The benefit of using one workspace per team is that all machine learning artifacts for the team's projects are stored in one place. You can see productivity increases because team members can easily access, explore, and reuse experimentation results. Organizing your workspaces by team reduces your Azure footprint and simplifies cost management by team. Because the number of experimentation assets can grow quickly, you can keep your artifacts organized by following naming and tagging conventions. For recommendations about how to name resources, see [Develop your naming and tagging strategy for Azure resources](#).

With this approach, each team member must have similar data access level permissions. Granular role-based access control (RBAC) and access control lists (ACL) for data sources and experimentation assets are limited within a workspace. You can't have use case data segregation requirements.

Workspace per project: Use one workspace for each project if you require segregation of data and experimentation assets by project, or have cost reporting and budgeting requirements at a project level. For example, you might have an organization with four machine learning teams that run three projects each for a total of 12 workspace instances.

The benefit of using one workspace per project is that you manage costs at the project level. A team typically creates a dedicated resource group for Azure Machine Learning and associated resources for similar reasons. When you work with external contributors, for example, a project-centered workspace simplifies collaboration on a project because external users only need to be granted access to the project resources, not the team resources.

Something to consider with this approach is the isolation of experimentation results and assets. The discovery and reuse of assets might be more difficult because assets are spread across multiple workspace instances.

Single Workspace: Use one workspace for non-team or non-project related work, or when costs can't be directly associated to a specific unit of billing, for example with R&D.

The benefit of this setup is the cost of individual, non-project related work can be decoupled from project-related costs. When you set up a single workspace for all users to do their individual work, you reduce your Azure footprint.

With this approach, the workspace might become cluttered quickly when many machine learning practitioners share the same instance. Users might require UI-based filtering of assets to effectively find their resources. You can create shared machine learning workspaces for each business division to mitigate scale concerns or to segment budgets.

Environments and workspace setup

An environment is a collection of resources that deployments target based on their stage in the application lifecycle. Common examples of environment names are Dev, Test, QA, Staging, and Production.

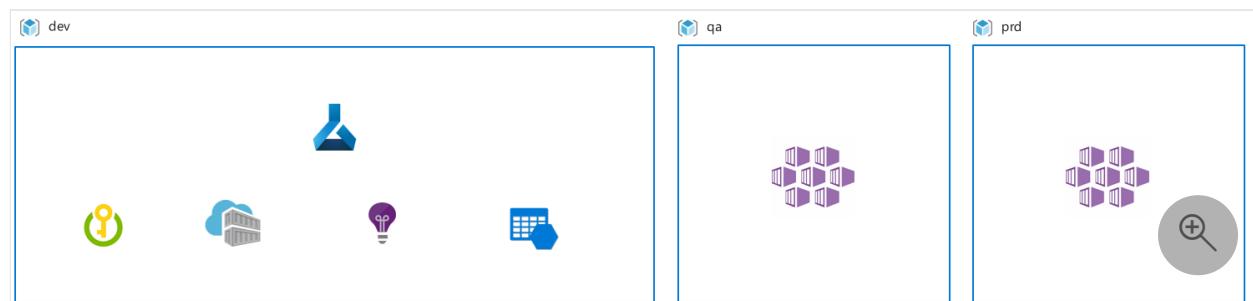
The development process in your organization affects requirements for environment usage. Your environment affects the setup of Azure Machine Learning and associated

resources, such as attached compute. For example, data availability might constrain the manageability of having a machine learning instance available for each environment. The following solution patterns are common:

Single environment workspace deployment: When you choose a single environment workspace deployment, Azure Machine Learning deploys to one environment. This setup is common for research-centered scenarios, where there's no need to release machine learning artifacts based on their lifecycle stage, across environments. Another scenario where this setup makes sense is when only inferencing services, and not machine learning pipelines, are deployed across environments.

The benefit of a research-centered setup is a smaller Azure footprint and minimal management overhead. This way of working implies no need to have an Azure Machine Learning workspace deployed in each environment.

With this approach, a single environment deployment is subject to data availability. So, be cautious when you set up your datastore. If you set up extensive access, for example, writer access on production data sources, you might unintentionally harm data quality. If you bring work to production in the same environment where the development happens, the same RBAC restrictions apply for both the development work and the production work. This setup might make both environments too rigid or too flexible.



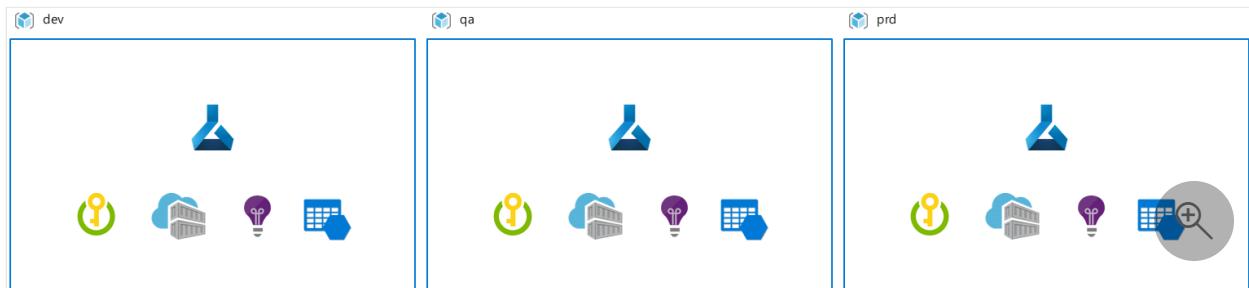
Multiple environment workspace deployment: When you choose a multiple environment workspace deployment, a workspace instance deploys for each environment. A common scenario for this setup is a regulated workplace with a clear separation of duties between environments, and for users who have resource access to those environments.

The benefits of this setup are:

- Staged rollout of machine learning workflows and artifacts. For example, models across environments, with the potential to enhance agility and reduce time-to-deployment.
- Enhanced security and control of resources because you can assign more access restrictions in downstream environments.

- Training scenarios on production data in non-development environments because you can give a select group of users access.

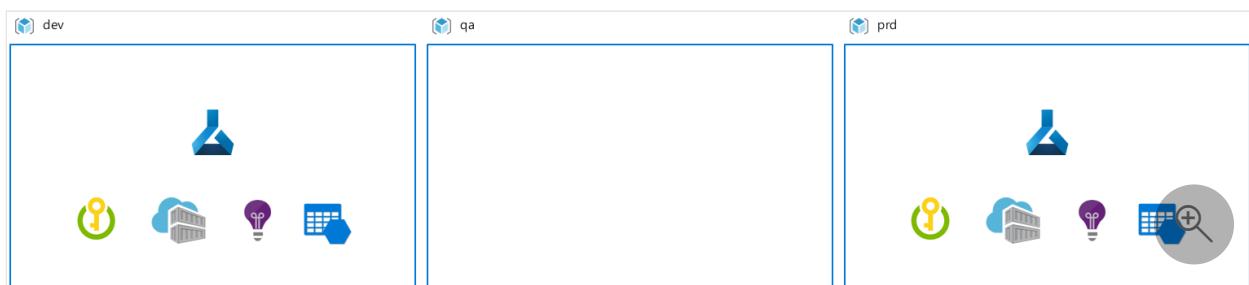
With this approach, you're at risk for more management and process overhead. This setup requires a fine-grained development and rollout process for machine learning artifacts across workspace instances. Also, data management and engineering effort might be required to make production data available for training in the development environment. Access management requires you to give a team access to resolve and investigate incidents in production. And finally, your team needs Azure DevOps and machine learning engineering expertise to implement automation workflows.



One environment with limited data access, one with production data access: When you choose this setup, Azure Machine Learning deploys to two environments: one with limited data access and one with production data access. This setup is common if you need to segregate development and production environments. For example, you might be working under organizational constraints to make production data available in any environment, or you might want to segregate development work from production work without duplicating data more than required due to the high cost of maintenance.

The benefit of this setup is the clear separation of duties and access between development and production environments. Another benefit is lower resource management overhead when compared to a multi-environment deployment scenario.

With this approach, you need a defined development and rollout process for machine learning artifacts across workspaces. Also, it might require data management and engineering effort to make production data available for training in a development environment. But this approach might require relatively less effort than a multi-environment workspace deployment.



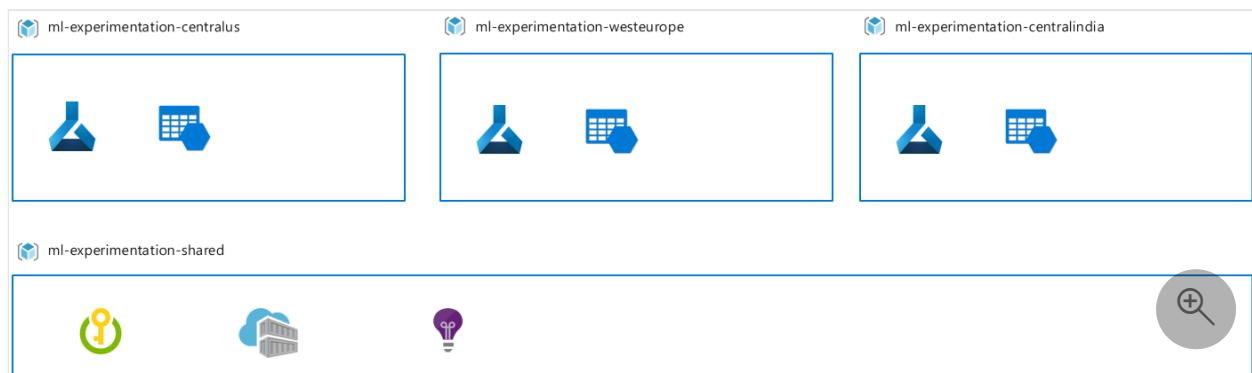
Regions and resource setup

The location of your resources, data, or users might require you to create Azure Machine Learning workspace instances and associated resources in multiple Azure regions. For example, one project might span its resources across the West Europe and East US Azure regions for performance, cost, and compliance reasons. The following scenarios are common:

Regional training: The machine learning training jobs run in the same Azure region as where the data is located. In this setup, a machine learning workspace deploys to each Azure region where data is located. This scenario is common when you need to meet compliance, or when you have data movement constraints across regions.

The benefit of this setup is you can do experimentation in the data center where the data is located with the least network latency. With this approach, when a machine learning pipeline runs across multiple workspace instances, it adds more management complexity. It becomes challenging to compare experimentation results across instances and adds overhead to quota and compute management.

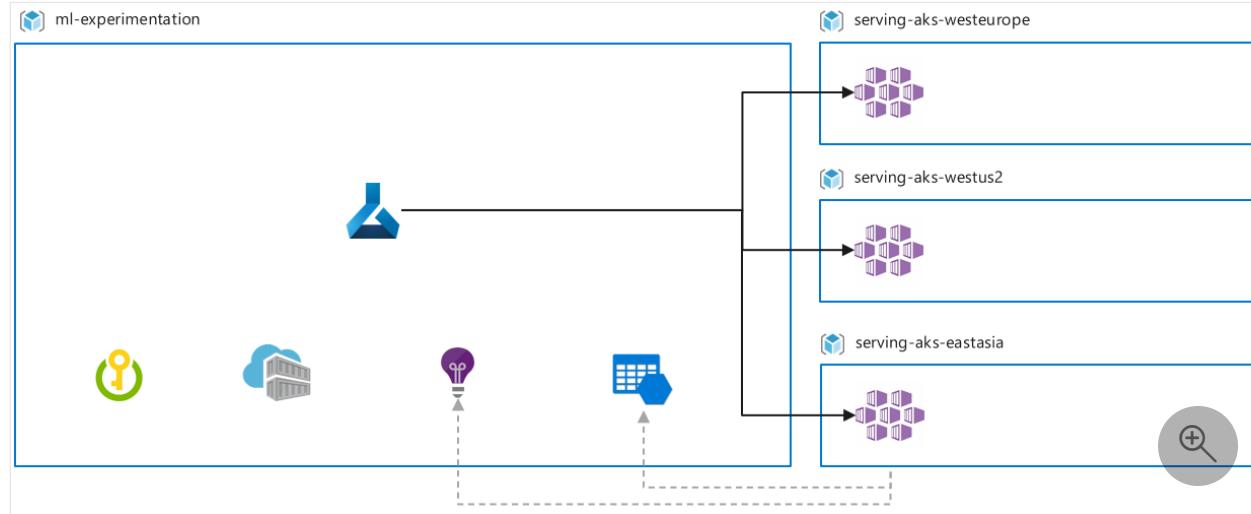
If you want to attach storage across regions, but use compute from one region, Azure Machine Learning supports the scenario of attaching storage accounts in a region rather than the workspace. Metadata, for example metrics, is stored in the workspace region.



Regional serving: Machine learning services deploy close to where the target audience lives. For example, if target users are in Australia and the main storage and experimentation region is West Europe, deploy the machine learning workspace for experimentation in West Europe. You then deploy an AKS cluster for inference endpoint deployment in Australia.

The benefits of this setup are the opportunity for inferencing in the data center where new data is ingested, minimizing latency and data movement, and compliance with local regulations.

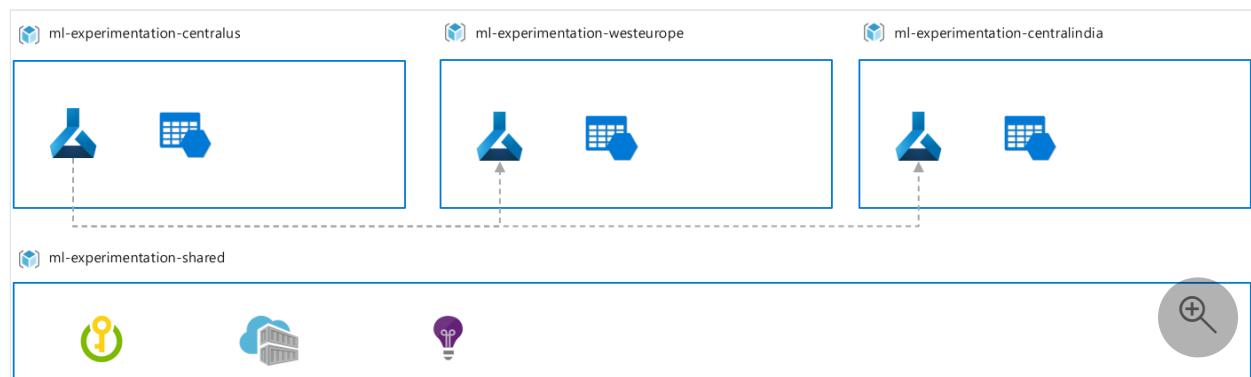
With this approach, a multi-region setup provides several advantages, but also adds more overhead on quota and compute management. When you have a requirement for batch inferencing, regional serving might require a multi-workspace deployment. Data collected through inferencing endpoints might need to be transferred across regions for retraining scenarios.



Regional fine-tuning: A base model trains on an initial dataset, for example, public data or data from all regions, and is later fine-tuned with a regional dataset. The regional dataset might only exist in a particular region because of compliance or data movement constraints. For example, you might need base model training to be done in a workspace in region A, while fine tuning happens in a workspace in region B.

The benefit of this setup is you can experiment compliantly in the data center where the data resides. You can also still take advantage of base model training on a larger dataset in an earlier pipeline stage.

This approach supports complex experimentation pipelines but it might create more challenges. For example, when you compare experiment results across regions, it might add more overhead to the quota and compute management.



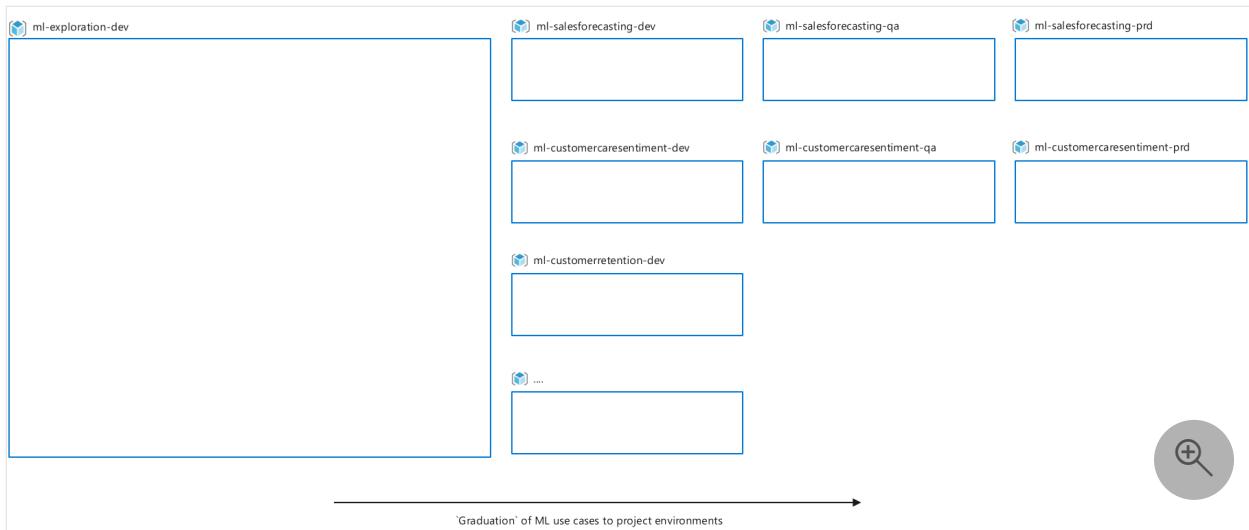
Reference implementation

To illustrate the deployment of Azure Machine Learning in a larger setting, this section shows how the organization 'Contoso' sets up Azure Machine Learning, given their organizational constraints, reporting, and budgeting requirements:

- Contoso creates resource groups on a solution basis for cost management and reporting reasons.
- IT administrators only create resource groups and resources for funded solutions to meet budget requirements.
- Because of the exploratory and uncertain nature of Data Science, users need a place to experiment and work for use case and data exploration. Often, exploratory work can't be directly associated to a particular use case, and can be associated only to an R&D budget. Contoso wants to fund some machine learning resources centrally that anyone can use for exploration purposes.
- Once a machine learning use case proves to be successful in the exploratory environment, teams can request resource groups. For example, the company can set up Dev, QA, and Production for iterative experimentation project work, and access to production data sources.
- Data segregation and compliance requirements don't allow live production data to exist in development environments.
- Different RBAC requirements exist for various user groups by IT policy per environment, for example, access is more restrictive in production.
- All data, experimentation, and inferencing happens in a single Azure region.

To adhere to the above requirements, Contoso sets up their resources in the following way:

- Azure Machine Learning workspaces and resource groups scoped per project to follow budgeting and use case segregation requirements.
- A multiple-environment setup for Azure Machine Learning and associated resources to address cost management, RBAC, and data access requirements.
- A single resource group and machine learning workspace that's dedicated for exploration.
- Azure Active Directory groups that are different per user role and environment. For example, operations that a data scientist can do in a production environment are different than in the development environment, and access levels might differ per solution.
- All resources created in a single Azure region.



Next steps

Learn about best practices on machine learning DevOps with Azure Machine Learning.

[Machine learning DevOps guide](#)

Learn about considerations when managing budgets, quota, and cost with Azure Machine Learning.

[Manage budgets, costs, and quota for Azure Machine Learning at organizational scale](#)

Azure Machine Learning best practices for enterprise security

Article • 10/18/2023

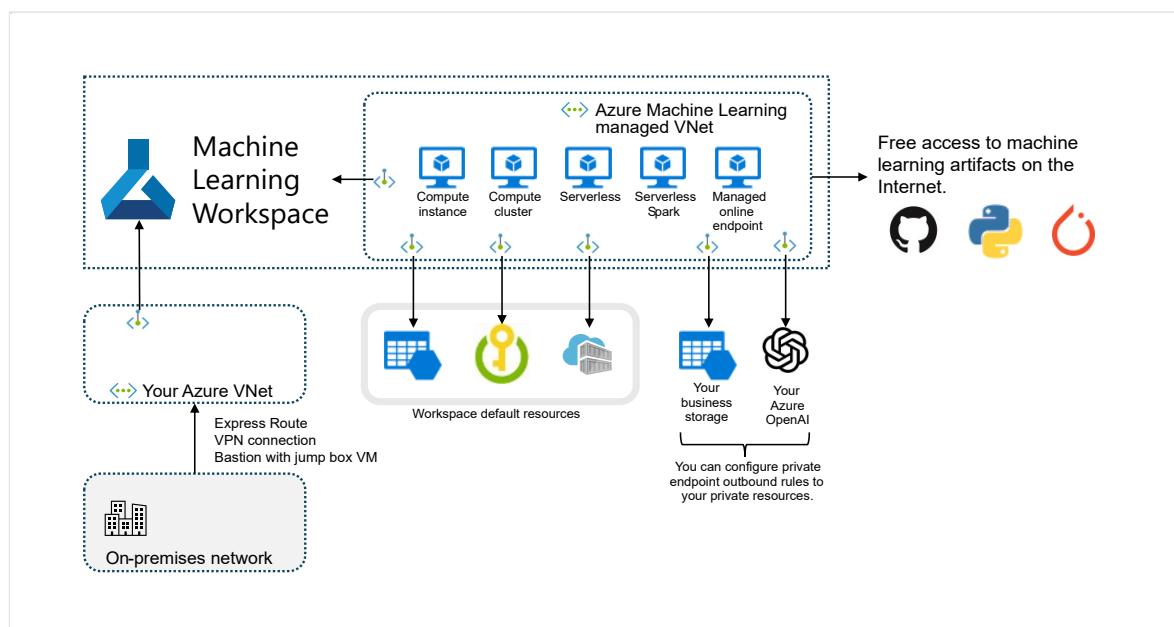
This article explains security best practices for planning or managing a secure Azure Machine Learning deployment. Best practices come from Microsoft and customer experience with Azure Machine Learning. Each guideline explains the practice and its rationale. The article also provides links to how-to and reference documentation.

Recommended network security architecture (managed network)

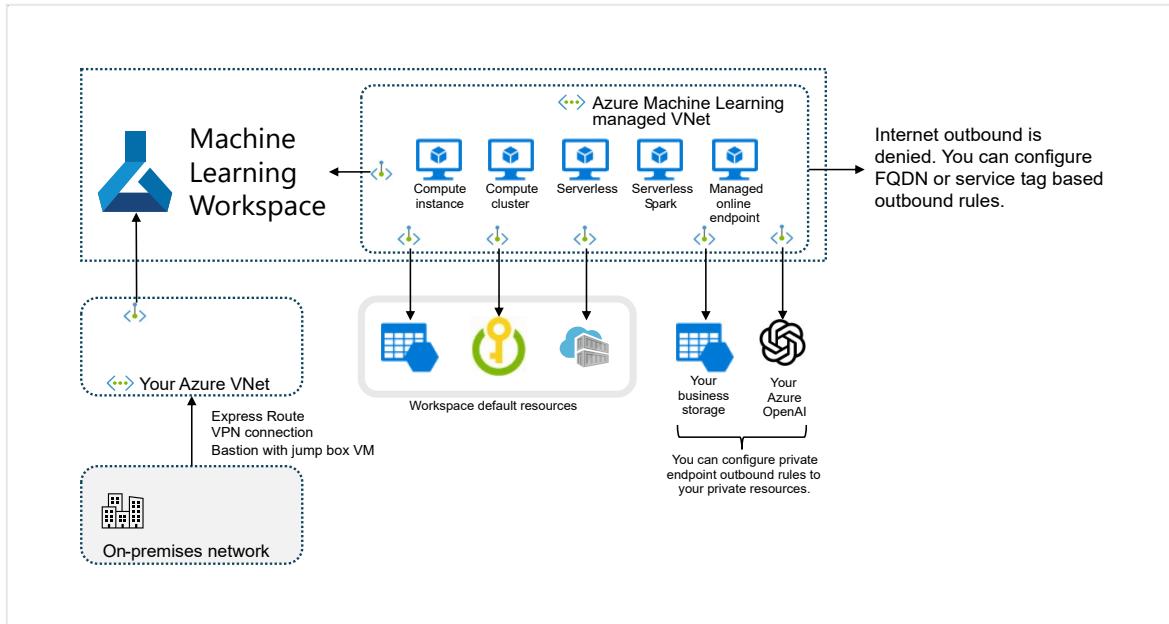
The recommended machine learning network security architecture is a *managed virtual network* (preview). An Azure Machine Learning managed virtual network secures the workspace, associated Azure resources, and all managed compute resources. It simplifies the configuration and management of network security by preconfiguring required outputs and automatically creating managed resources within the network. You can use private endpoints to allow Azure services to access the network and can optionally define outbound rules to allow the network to access the internet.

The managed virtual network has two modes that it can be configured for:

- **Allow internet outbound** - This mode allows outbound communication with resources located on the internet, such as the public PyPi or Anaconda package repositories.



- **Allow only approved outbound** - This mode allows only the minimum outbound communication required for the workspace to function. This mode is recommended for workspaces that must be isolated from the internet. Or where outbound access is only allowed to specific resources via service endpoints, service tags, or fully qualified domain names.

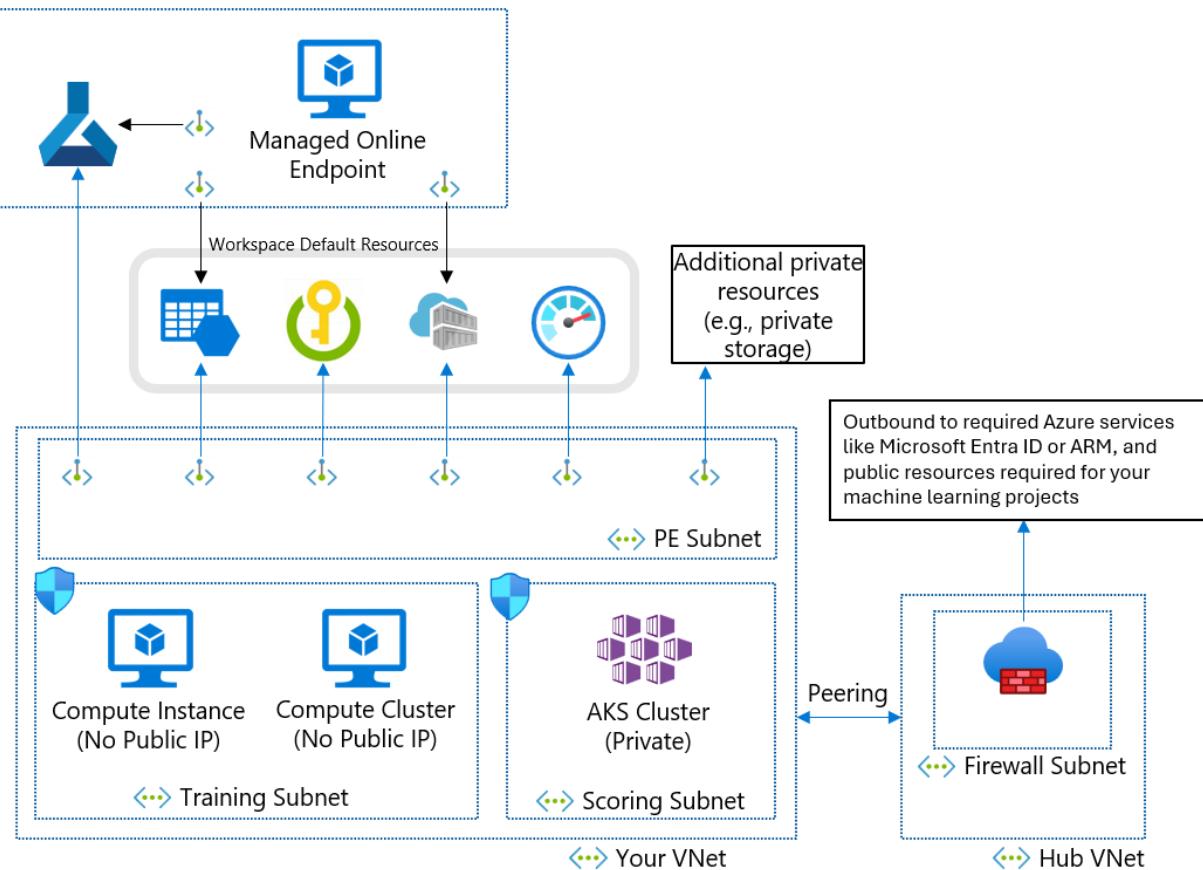


For more information, see [Managed virtual network isolation](#).

Recommended network security architecture (Azure Virtual Network)

If you can't use a managed virtual network due to your business requirements, you can use an Azure virtual network with the following subnets:

- **Training** contains compute resources used for training, such as machine learning compute instances or compute clusters.
- **Scoring** contains compute resources used for scoring, such as Azure Kubernetes Service (AKS).
- **Firewall** contains the firewall that allows traffic to and from the public internet, such as Azure Firewall.



The virtual network also contains a *private endpoint* for your machine learning workspace and the following dependent services:

- Azure Storage account
- Azure Key Vault
- Azure Container Registry

Outbound communication from the virtual network must be able to reach the following Microsoft services:

- Machine learning
- Microsoft Entra ID
- Azure Container Registry, and specific registries that Microsoft maintains
- Azure Front Door
- Azure Resource Manager
- Azure Storage

Remote clients connect to the virtual network using Azure ExpressRoute or a virtual private network (VPN) connection.

Virtual network and private endpoint design

When designing an Azure Virtual Network, subnets, and private endpoints, consider the following requirements:

- In general, create separate subnets for training and scoring and use the training subnet for all private endpoints.
- For IP addressing, compute instances need one private IP each. Compute clusters need one private IP per node. AKS clusters need many private IP addresses, as described in [Plan IP addressing for your AKS cluster](#). A separate subnet for at least AKS helps prevent IP address exhaustion.
- The compute resources in the training and scoring subnets must access the storage account, the key vault, and the container registry. Create private endpoints for the storage account, the key vault, and the container registry.
- Machine learning workspace default storage needs two private endpoints, one for Azure Blob Storage and another for Azure File Storage.
- If you use Azure Machine Learning studio, the workspace and storage private endpoints should be in the same virtual network.
- If you have multiple workspaces, use a virtual network for each workspace to create an explicit network boundary between workspaces.

Use private IP addresses

Private IP addresses minimize your Azure resources' exposure to the internet. Machine learning uses many Azure resources, and the machine learning workspace private endpoint isn't enough for end-to-end private IP. The following table shows the major resources machine learning uses and how to enable private IP for the resources.

Compute instances and compute clusters are the only resources that don't have the private IP feature.

Resources	Private IP solution	Documentation
Workspace	Private endpoint	Configure a private endpoint for an Azure Machine Learning workspace
Registry	Private endpoint	Network isolation with Azure Machine Learning registries
Associated resources		
Storage	Private endpoint	Secure Azure Storage accounts with service

Resources	Private IP solution	Documentation
		endpoints
Key Vault	Private endpoint	Secure Azure Key Vault
Container Registry	Private endpoint	Enable Azure Container Registry
Training resources		
Compute instance	Private IP (no public IP)	Secure training environments
Compute cluster	Private IP (no public IP)	Secure training environments
Hosting resources		
Managed online endpoint	Private endpoint	Network isolation with managed online endpoints
Online endpoint (Kubernetes)	Private endpoint	Secure Azure Kubernetes Service online endpoints
Batch endpoints	Private IP (inherited from compute cluster)	Network isolation in batch endpoints

Control virtual network inbound and outbound traffic

Use a firewall or Azure network security group (NSG) to control virtual network inbound and outbound traffic. For more information on inbound and outbound requirements, see [Configure inbound and outbound network traffic](#). For more information on traffic flows between components, see [Network traffic flow in a secured workspace](#).

Ensure access to your workspace

To ensure that your private endpoint can access your machine learning workspace, take the following steps:

1. Make sure you have access to your virtual network using a VPN connection, ExpressRoute, or jump box virtual machine (VM) with Azure Bastion access. The public user can't access the machine learning workspace with the private endpoint, because it can be accessed only from your virtual network. For more information, see [Secure your workspace with virtual networks](#).
2. Make sure you can resolve the workspace fully qualified domain names (FQDNs) with your private IP address. If you use your own Domain Name System (DNS) server or a [centralized DNS infrastructure](#), you need to configure a DNS forwarder. For more information, see [How to use your workspace with a custom DNS server](#).

Workspace access management

When defining machine learning identity and access management controls, you can separate controls that define access to Azure resources from controls that manage access to data assets. Depending on your use case, consider whether to use *self-service*, *data-centric*, or *project-centric* identity and access management.

Self-service pattern

In a self-service pattern, data scientists can create and manage workspaces. This pattern is best suited for proof-of-concept situations requiring flexibility to try different configurations. The disadvantage is that data scientists need the expertise to provision Azure resources. This approach is less suitable when strict control, resource use, audit traces, and data access are required.

1. Define Azure policies to set safeguards for resource provisioning and usage, such as allowed cluster sizes and VM types.
2. Create a resource group for holding the workspaces and grant data scientists a Contributor role in the resource group.
3. Data scientists can now create workspaces and associate resources in the resource group in a self-service manner.
4. To access data storage, create user-assigned managed identities and grant the identities read-access roles on the storage.
5. When data scientists create compute resources, they can assign the managed identities to the compute instances to gain data access.

For best practices, see [Authentication for cloud-scale analytics](#).

Data-centric pattern

In a data-centric pattern, the workspace belongs to a single data scientist who might be working on multiple projects. The advantage of this approach is that the data scientist can reuse code or training pipelines across projects. As long as the workspace is limited to a single user, data access can be traced back to that user when auditing storage logs.

The disadvantage is that data access isn't compartmentalized or restricted on a per-project basis, and any user added to the workspace can access the same assets.

1. Create the workspace.

2. Create compute resources with system-assigned managed identities enabled.
3. When a data scientist needs access to the data for a given project, grant the compute managed identity read access to the data.
4. Grant the compute managed identity access to other required resources, such as a container registry with custom Docker images for training.
5. Also grant the workspace's managed identity read-access role on the data to enable data preview.
6. Grant the data scientist access to the workspace.
7. The data scientist can now create data stores to access data required for projects and submit training runs that use the data.

Optionally, create a Microsoft Entra security group and grant it read access to data, then add managed identities to the security group. This approach reduces the number of direct role assignments on resources, to avoid reaching the subscription limit on role assignments.

Project-centric pattern

A project-centric pattern creates a machine learning workspace for a specific project, and many data scientists collaborate within the same workspace. Data access is restricted to the specific project, making the approach well suited for working with sensitive data. Also, it's straightforward to add or remove data scientists from the project.

The disadvantage of this approach is that sharing assets across projects can be difficult. It's also hard to trace data access to specific users during audits.

1. Create the workspace
2. Identify data storage instances required for the project, create a user-assigned managed identity, and grant the identity read access to the storage.

Optionally, grant the workspace's managed identity access to data storage to allow data preview. You can omit this access for sensitive data not suitable for preview.
3. Create credentialless data stores for the storage resources.
4. Create compute resources within the workspace, and assign the managed identity to the compute resources.

5. Grant the compute managed identity access to other required resources, such as a container registry with custom Docker images for training.
6. Grant data scientists working on the project a role on the workspace.

By using Azure role-based access control (RBAC), you can restrict data scientists from creating new datastores or new compute resources with different managed identities. This practice prevents access to data not specific to the project.

Optionally, to simplify project membership management, you can create a Microsoft Entra security group for project members and grant the group access to the workspace.

Azure Data Lake Storage with credential passthrough

You can use Microsoft Entra user identity for interactive storage access from machine learning studio. Data Lake Storage with hierarchical namespace enabled allows for enhanced organization of data assets for storage and collaboration. With Data Lake Storage hierarchical namespace, you can compartmentalize data access by giving different users access control list (ACL)-based access to different folders and files. For example, you can grant only a subset of users access to confidential data.

RBAC and custom roles

Azure RBAC helps you manage who has access to machine learning resources and configure who can perform operations. For example, you might want to grant only specific users the workspace administrator role to manage compute resources.

Access scope can differ between environments. In a production environment, you might want to limit the ability of users to update inference endpoints. Instead, you might grant that permission to an authorized service principal.

Machine learning has several default roles: owner, contributor, reader, and data scientist. You can also create your own custom roles, for example to create permissions that reflect your organizational structure. For more information, see [Manage access to Azure Machine Learning workspace](#).

Over time, the composition of your team might change. If you create a Microsoft Entra group for each team role and workspace, you can assign an Azure RBAC role to the Microsoft Entra group, and manage resource access and user groups separately.

User principals and service principals can be part of the same Microsoft Entra group. For example, when you create a user-assigned managed identity that Azure Data Factory

uses to trigger a machine learning pipeline, you might include the managed identity in a **ML pipelines executor** Microsoft Entra group.

Central Docker image management

Azure Machine Learning provides curated Docker images that you can use for training and deployment. However, your enterprise compliance requirements might mandate using images from a private repository your company manages. Machine learning has two ways to use a central repository:

- Use the images from a central repository as base images. The machine learning environment management installs packages and creates a Python environment where the training or inferencing code runs. With this approach, you can update package dependencies easily without modifying the base image.
- Use the images as-is, without using machine learning environment management. This approach gives you a higher degree of control but also requires you to carefully construct the Python environment as part of the image. You need to meet all the necessary dependencies to run the code, and any new dependencies require rebuilding the image.

For more information, see [Manage environments](#).

Data encryption

Machine learning data at rest has two data sources:

- Your storage has all your data, including training and trained model data, except for the metadata. You're responsible for your storage encryption.
- Azure Cosmos DB contains your metadata, including run history information like experiment name and experiment submission date and time. In most workspaces, Azure Cosmos DB is in the Microsoft subscription and encrypted by a Microsoft-managed key.

If you want to encrypt your metadata using your own key, you can use a customer-managed key workspace. The downside is that you need to have Azure Cosmos DB in your subscription and pay its cost. For more information, see [Data encryption with Azure Machine Learning](#).

For information on how Azure Machine Learning encrypts data in transit, see [Encryption in transit](#).

Monitoring

When you deploy machine learning resources, set up logging and auditing controls for observability. Motivations for observing data might vary based on who looks at the data. Scenarios include:

- Machine learning practitioners or operations teams want to **monitor machine learning pipeline health**. These observers need to understand issues in scheduled execution or problems with data quality or expected training performance. You can build Azure dashboards that [monitor Azure Machine Learning data](#) or [create event-driven workflows](#).
- Capacity managers, machine learning practitioners, or operations teams might want to [create a dashboard](#) to **observe compute and quota utilization**. To manage a deployment with multiple Azure Machine Learning workspaces, consider creating a central dashboard to understand quota utilization. Quotas are managed on a subscription level, so the environment-wide view is important to drive optimization.
- IT and operations teams can set up [diagnostic logging](#) to **audit resource access and altering events** in the workspace.
- Consider creating dashboards that **monitor overall infrastructure health** for machine learning and dependent resources such as storage. For example, combining Azure Storage metrics with pipeline execution data can help you optimize infrastructure for better performance or discover problem root causes.

Azure collects and stores platform metrics and activity logs automatically. You can route the data to other locations by using a diagnostic setting. Set up diagnostic logging to a centralized Log Analytics workspace for observability across several workspace instances. Use Azure Policy to automatically set up logging for new machine learning workspaces into this central Log Analytics workspace.

Azure Policy

You can enforce and audit the usage of security features on workspaces through Azure Policy. Recommendations include:

- Enforce custom-managed key encryption.
- Enforce Azure Private Link and private endpoints.
- Enforce private DNS zones.
- Disable non-Azure AD authentication, such as Secure Shell (SSH).

For more information, see [Built-in policy definitions for Azure Machine Learning](#).

You can also use custom policy definitions to govern workspace security in a flexible manner.

Compute clusters and instances

The following considerations and recommendations apply to machine learning compute clusters and instances.

Disk encryption

The operating system (OS) disk for a compute instance or compute cluster node is stored in Azure Storage and encrypted with Microsoft-managed keys. Each node also has a local temporary disk. The temporary disk is also encrypted with Microsoft-managed keys if the workspace was created with the `hbi_workspace = True` parameter. For more information, see [Data encryption with Azure Machine Learning](#).

Managed identity

Compute clusters support using managed identities to authenticate to Azure resources. Using a managed identity for the cluster allows authentication to resources without exposing credentials in your code. For more information, see [Create an Azure Machine Learning compute cluster](#).

Setup script

You can use a setup script to automate the customization and configuration of compute instances at creation. As an administrator, you can write a customization script to use when creating all compute instances in a workspace. You can use Azure Policy to enforce the use of the setup script to create every compute instance. For more information, see [Create and manage an Azure Machine Learning compute instance](#).

Create on behalf of

If you don't want data scientists to provision compute resources, you can create compute instances on their behalf and assign them to the data scientists. For more information, see [Create and manage an Azure Machine Learning compute instance](#).

Private endpoint-enabled workspace

Use compute instances with a private endpoint-enabled workspace. The compute instance rejects all public access from outside the virtual network. This configuration also prevents packet filtering.

Azure Policy support

When using an *Azure virtual network*, you can use Azure Policy to ensure that every compute cluster or instance is created in a virtual network and specify the default virtual network and subnet. The policy isn't needed when using a *managed virtual network*, as the compute resources are automatically created in the managed virtual network.

You can also use a policy to disable non-Azure AD authentication, such as SSH.

Next steps

Learn more about machine learning security configurations:

- [Enterprise security and governance](#)
- [Secure workspace resources using virtual networks](#)

Get started with a machine learning template-based deployment:

- [Azure Quickstart Templates \(microsoft.com\)](#)
- [Enterprise-scale analytics and AI data landing zone](#)

Read more articles about architectural considerations for deploying machine learning:

- Learn how team structure, environment, or regional constraints affect workspace setup.

[Organize and set up Azure Machine Learning environments](#)

- See how to manage compute costs and budget across teams and users.

[Budget, cost, and quota management for Azure Machine Learning at organizational scale](#)

- Learn about machine learning DevOps (MLOps), which uses a combination of people, process, and technology to deliver robust, reliable, and automated machine learning solutions.

[Machine learning DevOps guide](#)

Manage budgets, costs, and quota for Azure Machine Learning at organizational scale

Article • 01/26/2023

When you manage compute costs incurred from Azure Machine Learning, at an organization scale with many workloads, many teams, and users, there are numerous management and optimization challenges to work through.

In this article, we present best practices to optimize costs, manage budgets, and share quota with Azure Machine Learning. It reflects the experience and lessons learned from running machine learning teams internally at Microsoft and while partnering with our customers. You'll learn how to:

- [Optimize compute resources to meet workload requirements.](#)
- [Drive the best use of a team's budget.](#)
- [Plan, manage and share budgets, cost, and quota at enterprise-scale.](#)

Optimize compute to meet workload requirements

When you start a new machine learning project, exploratory work might be needed to get a good picture of compute requirements. This section provides recommendations on how you can determine the right virtual machine (VM) SKU choice for training, for inferencing, or as a workstation to work from.

Determine the compute size for training

Hardware requirements for your training workload might vary from project to project. To meet these requirements, Azure Machine Learning compute [offers various types](#) of VMs:

- **General purpose:** Balanced CPU to memory ratio.
- **Memory optimized:** High memory to CPU ratio.
- **Compute optimized:** High CPU to memory ratio.
- **High performance compute:** Deliver leadership-class performance, scalability, and cost efficiency for various real-world HPC workloads.
- **Instances with GPUs:** Specialized virtual machines targeted for heavy graphic rendering and video editing, as well as model training and inferencing (ND) with

deep learning.

You might not know yet what your compute requirements are. In this scenario, we recommend starting with either of the following cost effective default options. These options are for lightweight testing and for training workloads.

Type	Virtual machine size	Specs
CPU	Standard_DS3_v2	4 cores, 14 gigabytes (GB) RAM, 28-GB storage
GPU	Standard_NC6	6 cores, 56 gigabytes (GB) RAM, 380-GB storage, NVIDIA Tesla K80 GPU

To get the best VM size for your scenario, it might consist of trial and error. Here are several aspects to consider.

- If you need a CPU:
 - Use a [memory optimized](#) VM if you're training on large datasets.
 - Use a [compute optimized](#) VM if you're doing real-time inferencing or other latency sensitive tasks.
 - Use a VM with more cores and RAM in order to speed up training times.
- If you need a GPU, see the [GPU optimized VM sizes](#) for information on selecting a VM.
 - If you're doing distributed training, use VM sizes that have multiple GPUs.
 - If you're doing distributed training on multiple nodes, use GPUs that have NVLink connections.

While you select the VM type and SKU that best fits your workload, evaluate comparable VM SKUs as a trade-off between CPU and GPU performance and pricing. From a cost management perspective, a job might run reasonably well on several SKUs.

Certain GPUs such as the NC family, particularly NC_Promo SKUs, have similar abilities to other GPUs such as low latency and ability to manage multiple computing workloads in parallel. They're available at discounted prices compared to some of the other GPUs. Considerately selecting VM SKUs to the workload might save cost significantly in the end.

A reminder on the importance for utilization is to sign up for a greater number of GPUs doesn't necessarily execute with faster results. Instead, make sure the GPUs are fully utilized. For example, double check the need for NVIDIA CUDA. While it might be required for high-performance GPU execution, your job might not take a dependency on it.

Determine the compute size for inference

Compute requirements for inference scenarios differ from training scenarios. Available options differ based on whether your scenario demands offline inference in batch or requires online inference in real time.

For real-time inference scenarios consider the following suggestions:

- Use [profiling capabilities](#) on your model with Azure Machine Learning to determine how much CPU and memory you need to allocate for the model when deploying it as a web service.
- If you're doing real-time inference but don't need high availability, deploy to [Azure Container Instances](#) (no SKU selection).
- If you're doing real-time inference but need high availability, deploy to [Azure Kubernetes Service](#).
 - If you're using traditional machine learning models and receive < 10 queries/second, start with a CPU SKU. F-series SKUs often work well.
 - If you're using deep learning models and receive > 10 queries/second, try a NVIDIA GPU SKU (NCasT4_v3 often works well) [with Triton](#).

For batch inference scenarios consider the following suggestions:

- When you use Azure Machine Learning pipelines for batch inferencing, follow the guidance in [Determine the compute size for training](#) to choose your initial VM size.
- Optimize cost and performance by scaling horizontally. One of the key methods of optimizing cost and performance is by parallelizing the workload with the help of [parallel run step](#) in Azure Machine Learning. This pipeline step allows you to use many smaller nodes to execute the task in parallel, which allows you to scale horizontally. There's an overhead for parallelization though. Depending on the workload and the degree of parallelism that can be achieved, a parallel run step may or may not be an option.

Determine the size for compute instance

For interactive development, Azure Machine Learning's compute instance is recommended. The compute instance (CI) offering brings single node compute that's bound to a single user and can be used as a cloud workstation.

Some organizations disallow the use of production data on local workstations, have enforced restrictions to the workstation environment, or restrict the installation of packages and dependencies in the corporate IT environment. A compute instance can be used as a workstation to overcome the limitation. It offers a secure environment with

production data access, and runs on images that come with popular packages and tools for data science pre-installed.

When compute instance is running, user is billed for VM compute, Standard Load Balancer (included lb/outbound rules, and data processed), OS disk (Premium SSD managed P10 disk), temp disk (the temp disk type depends on the VM size chosen), and public IP address. To save costs, we recommend users consider:

- Start and stop the compute instance when it's not in use.
- Work with a sample of your data on a compute instance and scale out to compute clusters to work with your full set of data
- Submit experimentation jobs in *local* compute target mode on the compute instance while developing or testing, or when you switch to shared compute capacity when you submit jobs at full scale. For example, many epochs, full set of data, and hyperparameter search.

If you stop the compute instance, it stops billing for VM compute hours, temp disk, and Standard Load Balancer data processed costs. Note user still pays for OS disk and Standard Load Balancer included lb/outbound rules even when compute instance is stopped. Any data saved on OS disk is persisted through stop and restarts.

Tune the chosen VM size by monitoring compute utilization

You can view information on your Azure Machine Learning compute usage and utilization via Azure Monitor. You can view details on model deployment and registration, quota details such as active and idle nodes, run details such as canceled and completed runs, and compute utilization for GPU and CPU utilization.

Based on the insights from the monitoring details, you can better plan or adjust your resource usage across the team. For example, if you notice many idle nodes over the past week, you can work with the corresponding workspace owners to update the compute cluster configuration to prevent this extra cost. Benefits of analyzing the utilization patterns can help with forecasting costs and budget improvements.

You can access these metrics directly from the Azure portal. Go to your Azure Machine Learning workspace, and select *Metrics* under the monitoring section on the left panel. Then, you can select details on what you would like to view, such as metrics, aggregation, and time period. For more information, see [Monitor Azure Machine Learning](#) documentation page.

The screenshot shows the Azure Machine Learning Metrics blade. On the left, there's a sidebar with sections like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Events, Settings, Properties, Locks, Monitoring, Alerts, Metrics (which is selected), Diagnostic settings, and Logs. The main area has tabs for New chart, Refresh, Share, Feedback, Line chart, Drill into Logs, New alert rule, Pin to dashboard, and Local Time: Last 24 hours (Automatic). A search bar at the top left says 'Search (Ctrl+)/'. Below the tabs, there are filters for Add metric, Add filter, and Apply splitting. The 'Scope' dropdown is set to 'brendal-test'. The 'Metric Namespace' dropdown is set to 'Machine Learning Serv...'. The 'Metric' dropdown is open, showing a list of metrics under 'MODEL': Model Deploy Failed, Model Deploy Started, Model Deploy Succeeded, Model Register Failed, Model Register Succeeded, and Active Cores. The 'Aggregation' dropdown is also open. The 'Local Time: Last 24 hours (Automatic)' button is visible at the top right.

Switch between local, single-node, and multi-node cloud compute while you develop

There are varying compute and tooling requirements throughout the machine learning lifecycle. Azure Machine Learning can be interfaced with through an SDK and CLI interface from practically any preferred workstation configuration to meet these requirements.

To save costs and work productively, it's recommended to:

- Clone your experimentation code base locally by using Git and submit jobs to cloud compute using the Azure Machine Learning SDK or CLI.
- If your dataset is large, consider managing a sample of your data on your local workstation, while keeping the full dataset on cloud storage.
- Parameterize your experimentation code base so that you can configure your jobs to run with a varying number of epochs or on datasets of different sizes.
- Don't hard code the folder path of your dataset. You can then easily reuse the same code base with different datasets, and under local and cloud execution context.
- Bootstrap your experimentation jobs in *local* compute target mode while you develop or test, or when you switch to a shared compute cluster capacity when you submit jobs at full scale.
- If your dataset is large, work with a sample of data on your local or compute instance workstation, while scaling to cloud compute in Azure Machine Learning to work with your full set of data.
- When your jobs take a long time to execute, consider optimizing your code base for distributed training to allow for scaling out horizontally.

- Design your distributed training workloads for node elasticity, to allow flexible use of single-node and multi-node compute, and ease usage of compute that can be preempted.

Combine compute types using Azure Machine Learning pipelines

When you orchestrate your machine learning workflows, you can define a pipeline with multiple steps. Each step in the pipeline can run on its own compute type. This allows you to optimize performance and cost to meet varying compute requirements across the machine learning lifecycle.

Drive the best use of a team's budget

While budget allocation decisions might be out of the span of control of an individual team, a team is typically empowered to use their allocated budget to their best needs. By trading off job priority versus performance and cost wisely, a team can achieve higher cluster utilization, lower overall cost, and use a larger number of compute hours from the same budget. This can result in enhanced team productivity.

Optimize the costs of shared compute resources

The key to optimize costs of shared compute resources is to ensure that they're being used to their full capacity. Here are some tips to optimize your shared resource costs:

- When you use compute instances, only turn them on when you have code to execute. Shut them down when they aren't being used.
- When you use compute clusters, set the minimum node count to 0 and the maximum node count to a number that is evaluated based on your budget constraints. Use the [Azure pricing calculator](#) to calculate the cost of full utilization of one VM node of your chosen VM SKU. Autoscaling will scale down all the compute nodes when there's no one using it. It will only scale up to the number of nodes you have budget for. You can configure [autoscaling](#) to scale down all the compute nodes.
- Monitor your resource utilizations such as CPU utilization and GPU utilization when training models. If the resources aren't being fully used, modify your code to better use resources or scale down to smaller or cheaper VM sizes.
- Evaluate whether you can create shared compute resources for your team to avoid computing inefficiencies caused by cluster scaling operations.
- Optimize compute cluster autoscaling timeout policies based on usage metrics.

- Use workspace quotas to control the amount of compute resources that individual workspaces have access to.

Introduce scheduling priority by creating clusters for multiple VM SKUs

Acting under quota and budget constraints, a team must trade off timely execution of jobs versus cost, to ensure important jobs run timely and a budget is used in the best way possible.

To support best compute utilization, teams are recommended to create clusters of various sizes and with *low priority* and *dedicated* VM priorities. Low-priority computes make use of surplus capacity in Azure and hence come with discounted rates. On the downside, these machines can be preempted anytime a higher priority ask comes in.

Using the clusters of varying size and priority, a notion of scheduling priority can be introduced. For example, when experimental and production jobs compete for the same NC GPU-quota, a production job might have preference to run over the experimental job. In that case, run the production job on the dedicated compute cluster, and the experimental job on the low priority compute cluster. When quota falls short, the experimental job will be preempted in favor of the production job.

Next to VM priority, consider running jobs on various VM SKUs. It might be that a job takes longer to execute on a VM instance with a P40 GPU than on a V100 GPU. However, since V100 VM instances might be occupied or quota fully used, the time to completion on the P40 might still be faster from a job throughput perspective. You might also consider running jobs with lower priority on less performant and cheaper VM instances from a cost management perspective.

Early-terminate a run when training doesn't converge

When you continuously experiment to improve a model against its baseline, you might be executing various experiment runs, each with slightly different configurations. For one run, you might tweak the input datasets. For another run, you might make a hyperparameter change. Not all changes might be as effective as the other. You detect early that a change didn't have the intended effect on the quality of your model training. To detect if training does not converge, monitor training progress during a run. For example, by logging performance metrics after each training epoch. Consider early terminating the job to free up resources and budget for another trial.

Plan, manage and share budgets, cost, and quota

As an organization grows its number of machine learning use cases and teams, it requires an increased operating maturity from IT and finance as well as coordination between individual machine learning teams to ensure efficient operations. Company-scale capacity and quota management become important to address scarceness of compute resources and overcome management overhead.

This section discusses best practices for planning, managing, and sharing budgets, cost, and quota at enterprise-scale. It's based on learnings from managing many GPU training resources for machine learning internally at Microsoft.

Understanding resource spend with Azure Machine Learning

One of the biggest challenges as an administrator for planning compute needs is starting new with no historical information as a baseline estimate. On a practical sense, most projects will start from a small budget as a first step.

To understand where the budget is going, it's critical to know where Azure Machine Learning costs come from:

- Azure Machine Learning only charges for compute infrastructure used and doesn't add a surcharge on compute costs.
- When an Azure Machine Learning workspace is created, there are also a few other resources created to enable Azure Machine Learning: Key Vault, Application Insights, Azure Storage, and Azure Container Registry. These resources are used in Azure Machine Learning and you'll pay for these resources.
- There are costs associated with managed compute such as training clusters, compute instances, and managed inferencing endpoints. With these managed compute resources, there are the following infrastructure costs to account for: virtual machines, virtual network, load balancer, bandwidth, and storage.

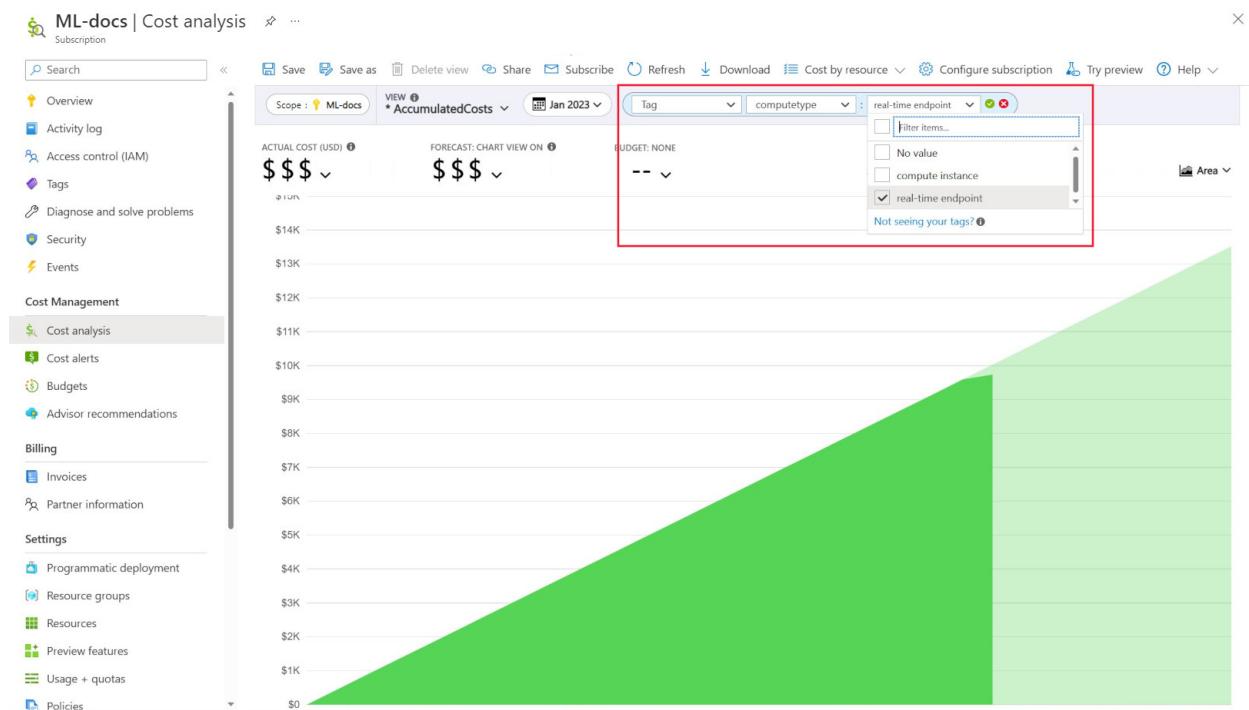
Track spending patterns and achieve better reporting with tagging

Administrators often want to be able to track costs on different resources in Azure Machine Learning. Tagging is a natural solution to this problem and aligns with the general approach used by Azure and many other cloud service providers. With tags

support, you can now see cost breakdown at the compute level, therefore granting you access to a more granular view to assist with better cost monitoring, improved reporting and greater transparency.

Tagging enables you to place customized tags on your workspaces and computes (from Azure Resource Manager templates and Azure Machine Learning studio) to further filter on these resources in Azure Cost Management based on these tags to observe spend patterns. This functionality can be best utilized for internal charge-back scenarios. In addition, tags can be useful for capturing metadata or details associated with the compute, for e.g. a project, a team, certain billing code, etc. This makes tagging very beneficial for measuring how much money you are spending on different resources and therefore, gaining deeper insights into your cost and spend patterns across teams or projects.

There are also system injected tags placed on computes that allow you to filter in the Cost Analysis page by the “Compute type” tag to see a compute wise breakdown of your total spend and determine what category of compute resources might be attributing to the majority of your costs. This is particularly useful for gaining more visibility into your training vs inferencing cost patterns.



Govern and restrict compute usage by policy

When you manage an Azure environment with many workloads, it can be a challenge to keep the overview on resource spend. [Azure Policy](#) can help control and govern resource spend, by restricting particular usage patterns across the Azure environment.

In specific for Azure Machine Learning, we recommend setting up policies to allow only for usage of specific VM SKUs. Policies can help prevent and control selection of expensive VMs. Policies can also be used to enforce usage of low-priority VM SKUs.

Allocate and manage quota based on business priority

Azure allows you to set limits for quota allocation on a subscription and Azure Machine Learning workspace level. Restricting who can manage quota through [Azure role-based access control \(RBAC\)](#) can help ensure resource utilization and cost predictability.

Availability of GPU quota can be scarce across your subscriptions. To ensure high quota utilization across workloads, we recommend monitoring whether quota is best used and assigned across workloads.

At Microsoft, it's determined periodically whether GPU quotas are best used and allocated across machine learning teams by evaluating capacity needs against business priority.

Commit capacity ahead of time

If you have a good estimate of how much compute will be used in the next year or next few years, you can purchase Azure Reserved VM Instances at a discounted cost. There are one-year or three-year purchase terms. Because Azure Reserved VM Instances are discounted, there can be significant cost savings compared to pay-as-you go prices.

Azure Machine Learning supports reserved compute instances. Discounts are automatically applied against Azure Machine Learning managed compute.

Manage data retention

Every time a machine learning pipeline is executed, intermediate datasets can be generated at each pipeline step for data caching and reuse. The growth of data as an output of these machine learning pipelines can become a pain point for an organization that is running many machine learning experiments.

Data scientists typically don't spend their time to clean up the intermediate datasets that are generated. Over time, the amount of data that is generated will add up. Azure Storage comes with a capability to enhance the management of the data lifecycle. Using [Azure Blob Storage lifecycle management](#), you can set up general policies to move data that is unused into colder storage tiers and save costs.

Infrastructure cost optimization considerations

Networking

Azure networking cost is incurred from outbound bandwidth from Azure datacenter. All inbound data to an Azure datacenter is free. The key to reduce network cost is to deploy all your resources in the same datacenter region whenever possible. If you can deploy Azure Machine Learning workspace and compute in the same region that has your data, you can enjoy lower cost and higher performance.

You might want to have private connection between your on-premises network and your Azure network to have a hybrid cloud environment. ExpressRoute enables you to do that but considering the high cost of ExpressRoute, it might be more cost effective to move away from a hybrid cloud setup and move all resources to Azure cloud.

Azure Container Registry

For Azure Container Registry, the determining factors for cost optimization include:

- Required throughput for Docker image downloads from the container registry to Azure Machine Learning
- Requirements for enterprise security features, such as Azure Private Link

For production scenarios where high throughput or enterprise security is required, the Premium SKU of Azure Container Registry is recommended.

For dev/test scenarios where throughput and security are less critical, we recommend either Standard SKU or Premium SKU.

The Basic SKU of Azure Container Registry isn't recommended for Azure Machine Learning. It's not recommended because of its low throughput and low included storage, which can be quickly exceeded by Azure Machine Learning's relatively large sized (1+ GB) Docker images.

Consider computing type availability when choosing Azure regions

When you [pick a region for your compute](#), keep the compute quota availability in mind. Popular and larger regions such as East US, West US, and West Europe tend to have higher default quota values and greater availability of most CPUs and GPUs, compared to some other regions with stricter capacity restrictions in place.

Learn more

Track costs across business units, environments, or projects by using the Cloud Adoption Framework

Next steps

To learn more about how to organize and set up Azure Machine Learning environments, see [Organize and set up Azure Machine Learning environments](#).

[Organize and set up Azure Machine Learning environments](#)

To learn about best practices on Machine Learning DevOps with Azure Machine Learning, see [Machine learning DevOps guide](#).

[Machine learning DevOps guide](#)

Machine learning operations

Article • 09/22/2022

Machine learning operations (also called *MLOps*) is the application of DevOps principles to AI-infused applications. To implement machine learning operations in an organization, specific skills, processes, and technology must be in place. The objective is to deliver machine learning solutions that are robust, scalable, reliable, and automated.

In this article, learn how to plan resources to support machine learning operations at the organization level. Review best practices and recommendations that are based on using Azure Machine Learning to adopt machine learning operations in the enterprise.

What is machine learning operations?

Modern machine learning algorithms and frameworks make it increasingly easier to develop models that can make accurate predictions. Machine learning operations is a structured way to incorporate machine learning in application development in the enterprise.

In an example scenario, you've built a machine learning model that exceeds all your accuracy expectations and impresses your business sponsors. Now it's time to deploy the model to production, but that might not be as easy as you had expected. The organization likely will need to have people, processes, and technology in place before it can use your machine learning model in production.

Over time, you or a colleague might develop a new model that works better than the original model. Replacing a machine learning model that's used in production introduces some concerns that are important to the organization:

- You'll want to implement the new model without disrupting the business operations that rely on the deployed model.
- For regulatory purposes, you might be required to explain the model's predictions or re-create the model if unusual or biased predictions result from data in the new model.
- The data you use in your machine learning training and model might change over time. With changes in the data, you might need to periodically retrain the model to maintain its prediction accuracy. A person or role will need to be assigned responsibility to feed the data, monitor the model's performance, retrain the model, and fix the model if it fails.

Suppose you have an application that serves a model's predictions via REST API. Even a simple use case like this one might cause problems in production. Implementing a machine learning operations strategy can help you address deployment concerns and support business operations that rely on AI-infused applications.

Some machine learning operations tasks fit well in the general DevOps framework. Examples include setting up unit tests and integration tests and tracking changes by using version control. Other tasks are more unique to machine learning operations and might include:

- Enable continuous experimentation and comparison against a baseline model.
- Monitor incoming data to detect [data drift](#).
- Trigger model retraining and set up a rollback for disaster recovery.
- Create reusable data pipelines for training and scoring.

The goal of machine learning operations is to close the gap between development and production and to deliver value to customers faster. To achieve this goal, you must rethink traditional development and production processes.

Not every organization's machine learning operations requirements are the same. The machine learning operations architecture of a large, multinational enterprise probably won't be the same infrastructure that a small startup establishes. Organizations typically begin small and build up as their maturity, model catalog, and experience grows.

The [machine learning operations maturity model](#) can help you see where your organization is on the machine learning operations maturity scale and help you plan for future growth.

Machine learning operations vs. DevOps

Machine learning operations is different from DevOps in several key areas. Machine learning operations has these characteristics:

- Exploration precedes development and operations.
- The data science lifecycle requires an adaptive way of working.
- Limits on data quality and availability limit progress.
- A greater operational effort is required than in DevOps.
- Work teams require specialists and domain experts.

For a summary, review the [seven principles of machine learning operations](#).

Exploration precedes development and operations

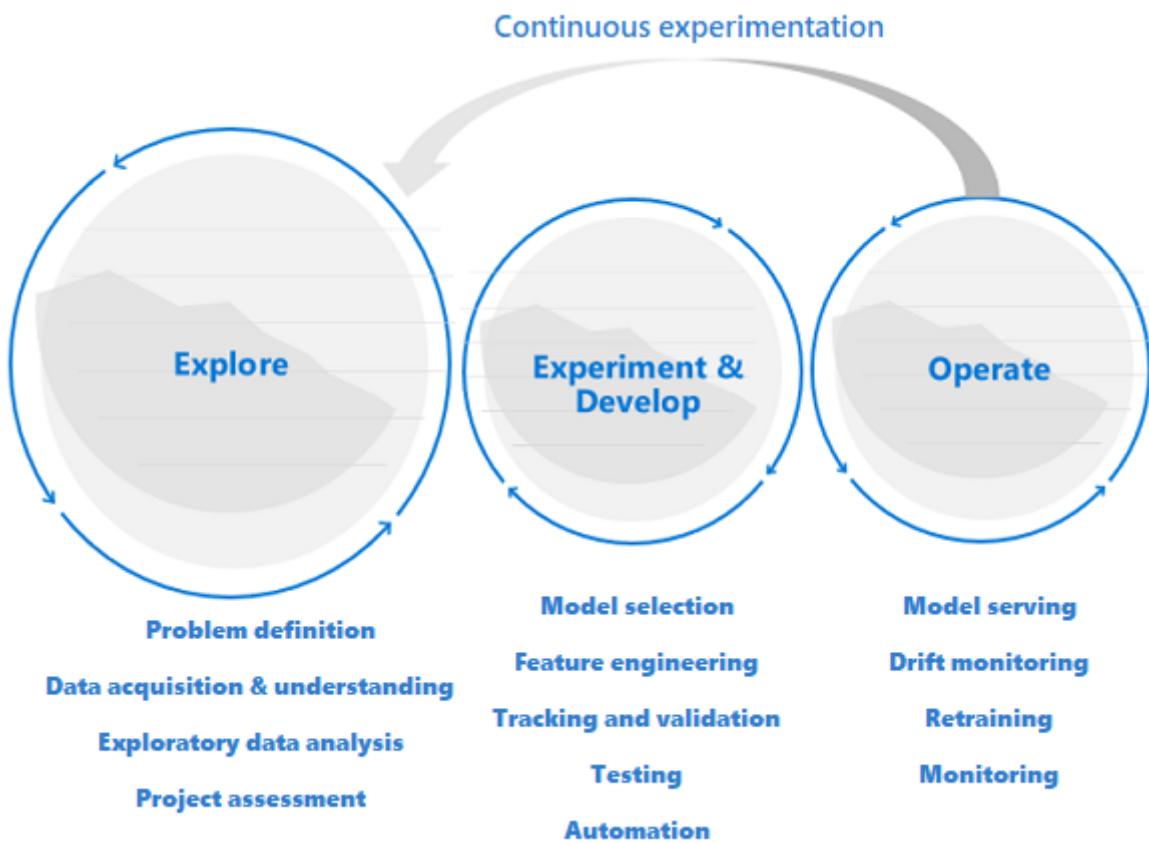
Data science projects are different from application development or data engineering projects. A data science project might make it to production, but often more steps are involved than in a traditional deployment. After an initial analysis, it might become clear that the business outcome can't be achieved with the available datasets. A more detailed exploration phase usually is the first step in a data science project.

The goal of the exploration phase is to define and refine the problem. During this phase, data scientists run exploratory data analysis. They use statistics and visualizations to confirm or falsify the problem hypotheses. Stakeholders should understand that the project might not extend beyond this phase. At the same time, it's important to make this phase as seamless as possible for a quick turnaround. Unless the problem to solve includes a security element, avoid restricting the exploratory phase with processes and procedures. Data scientists should be allowed to work with the tools and data they prefer. Real data is needed for this exploratory work.

The project can move to the experimentation and development stages when stakeholders are confident that the data science project is feasible and can provide real business value. At this stage, development practices become increasingly important. It's a good practice to capture metrics for all of the experiments that are done at this stage. It's also important to incorporate source control so that you can compare models and toggle between different versions of the code.

Development activities include refactoring, testing, and automating exploration code in repeatable experimentation pipelines. The organization must create applications and pipelines to serve the models. Refactoring code in modular components and libraries helps increase reusability, testing, and performance optimization.

Finally, the application or batch inference pipelines that serve the models are deployed to staging or production environments. In addition to monitoring infrastructure reliability and performance like for a standard application, in a machine learning model deployment, you must continuously monitor the quality of the data, the data profile, and the model for degradation or drift. Machine learning models also require retraining over time to stay relevant in a changing environment.



Data science lifecycle requires an adaptive way of working

Because the nature and quality of data initially is uncertain, you might not accomplish your business goals if you apply a typical DevOps process to a data science project. Exploration and experimentation are recurring activities and needs throughout the machine learning process. Teams at Microsoft use a project lifecycle and a working process that reflect the nature of data science-specific activities. The [Team Data Science Process](#) and The [Data Science Lifecycle Process](#) are examples of reference implementations.

Limits on data quality and availability limit progress

For a machine learning team to effectively develop machine learning-infused applications, access to production data is preferred for all relevant work environments. If production data access isn't possible due to compliance requirements or technical constraints, consider implementing [Azure role-based access control \(Azure RBAC\)](#) with [Azure Machine Learning](#), [just-in-time access](#), or [data movement pipelines](#) to create production data replicas and enhance user productivity.

Machine learning requires a greater operational effort

Unlike traditional software, the performance of a machine learning solution is constantly at risk because the solution is dependent on data quality. To maintain a qualitative solution in production, it's critical that you [continuously monitor and reevaluate both data and model quality](#). It's expected that a production model requires timely retraining, redeployment, and tuning. These tasks come on top of day-to-day security, [infrastructure monitoring](#), and compliance requirements, and they require specialized expertise.

Machine learning teams require specialists and domain experts

Although data science projects share roles with regular IT projects, the success of a machine learning effort highly depends on having essential machine learning technology specialists and domain subject matter experts. A technology specialist has the right background to do end-to-end machine learning experimentation. A domain expert can support the specialist by analyzing and synthesizing data or by qualifying data for use.

Common technical roles that are unique to data science projects are domain expert, data engineer, data scientist, AI engineer, model validator, and machine learning engineer. To learn more about roles and tasks in a typical data science team, see the [Team Data Science Process](#).

Seven principles of machine learning operations

As you plan to adopt machine learning operations in your organization, consider applying the following core principles as the foundation:

- **Use version control for code, data, and experimentation outputs.** Unlike in traditional software development, data has a direct influence on the quality of machine learning models. You should version your experimentation code base, but also version your datasets to ensure that you can reproduce experiments or inference results. Versioning experimentation outputs like models can save effort and the computational cost of re-creating them.
- **Use multiple environments.** To separate development and testing from production work, [replicate](#) your infrastructure in at least two environments. Access control for users might be different for each environment.
- **Manage your infrastructure and configurations as code.** When you create and update infrastructure components in your work environments, use [infrastructure as code](#), so inconsistencies don't develop in your environments. Manage machine

learning experiment job specifications as code so that you can easily rerun and reuse a version of your experiment in multiple environments.

- **Track and manage machine learning experiments.** Track key performance indicators and other artifacts for your machine learning experiments. When you keep a history of job performance, you can do a quantitative analysis of experimentation success and enhance team collaboration and agility.
- **Test code, validate data integrity, and ensure model quality.** [Test](#) your experimentation code base for correct data preparation and feature extraction functions, data integrity, and model performance.
- **Machine learning continuous integration and delivery.** Use [continuous integration \(CI\)](#) to automate testing for your team. Include model training as part of continuous training pipelines. Include A/B testing as part of your [release](#) to ensure that only a qualitative model is used in production.
- **Monitor services, models, and data.** When you serve models in a machine learning operations environment, it's critical to monitor the services for their infrastructure uptime, compliance, and model quality. [Set up monitoring](#) to identify data and model drift and to understand whether retraining is required. Consider setting up triggers for automatic retraining.

Best practices from Azure Machine Learning

Azure Machine Learning offers asset management, orchestration, and automation services to help you manage the lifecycle of your machine learning model training and deployment workflows. Review the best practices and recommendations to apply machine learning operations in the resource areas of people, process, and technology, all supported by Azure Machine Learning.

People

- Work in project teams to best use specialist and domain knowledge in your organization. Set up [Azure Machine Learning workspaces](#) for each project to comply with use case segregation requirements.
- Define a set of responsibilities and tasks as a role so that any team member on a machine learning operations project team can be assigned to and fulfill multiple roles. Use custom roles in Azure to define a set of granular [Azure RBAC operations for Azure Machine Learning](#) that each role can perform.

- Standardize on a project lifecycle and Agile methodology. The [Team Data Science Process](#) provides a reference lifecycle implementation.
- Balanced teams can run all machine learning operations stages, including exploration, development, and operations.

Process

- Standardize on a code template for code reuse and to accelerate ramp-up time on a new project or when a new team member joins the project. Use [Azure Machine Learning pipelines](#), [job submission scripts](#), and [CI/CD pipelines](#) as a basis for new templates.
- Use version control. Jobs that are submitted from a Git-backed folder [automatically track repo metadata](#) with the job in Azure Machine Learning for reproducibility.
- Use versioning for experiment inputs and outputs for reproducibility. Use [Azure Machine Learning datasets](#), [model management](#), and [environment management](#) capabilities to facilitate versioning.
- Build up a [run history](#) of experiment runs for comparison, planning, and collaboration. Use an experiment-tracking framework like [MLflow](#) to collect metrics.
- Continuously measure and control the quality of your team's work through [CI](#) on the full experimentation code base.
- Terminate training early in the process when a model doesn't converge. Use an experiment-tracking framework and the [run history](#) in Azure Machine Learning to monitor job runs.
- Define an experiment and model management strategy. Consider using a name like *champion* to refer to the current baseline model. A *challenger* model is a candidate model that might outperform the *champion* model in production. Apply tags in Azure Machine Learning to mark experiments and models. In a scenario like sales forecasting, it might take months to determine whether the model's predictions are accurate.
- Elevate [CI](#) for continuous training by including model training in the build. For example, begin model training on the full dataset with each pull request.
- Shorten the time it takes to get feedback on the quality of the machine learning pipeline by running an automated build on a data sample. Use [Azure Machine](#)

Learning pipeline parameters to parameterize input [datasets](#).

- Use [continuous deployment \(CD\) for machine learning models](#) to automate deployment and testing real-time scoring services in your Azure environments.
- In some regulated industries, you might be required to complete model validation steps before you can use a machine learning model in a production environment. Automating validation steps might accelerate time to delivery. When manual review or validation steps are still a bottleneck, consider whether you can certify the automated model validation pipeline. Use resource tags in Azure Machine Learning to indicate asset compliance and candidates for review or as triggers for deployment.
- Don't retrain in production, and then directly replace the production model without doing integration testing. Even though model performance and functional requirements might appear good, among other potential issues, a retrained model might have a larger environment footprint and break the server environment.
- When production data access is available only in production, use [Azure RBAC](#) and [custom roles](#) to give a select number of machine learning practitioners read access. Some roles might need to read the data for related data exploration. Alternatively, make a data copy available in nonproduction environments.
- Agree on naming conventions and tags for Azure Machine Learning [experiments](#) to differentiate retraining baseline machine learning pipelines from experimental work.

Technology

- If you currently submit jobs via the Azure Machine Learning studio UI or CLI, instead of submitting jobs via the SDK, use the CLI or [Azure DevOps Machine Learning tasks](#) to configure automation pipeline steps. This process might reduce the code footprint by reusing the same job submissions directly from automation pipelines.
- Use event-based programming. For example, trigger an offline model testing pipeline by using Azure Functions after a new model is registered. Or, send a notification to a designated email alias when a critical pipeline fails to run. Azure Machine Learning [creates events in Azure Event Grid](#). Multiple roles can subscribe to be notified of an event.
- When you use Azure DevOps for automation, use [Azure DevOps Tasks for Machine Learning](#) to use machine learning models as pipeline triggers.

- When you develop Python packages for your machine learning application, you can host them in an Azure DevOps repository as artifacts and publish them as a feed. By using this approach, you can [integrate](#) the DevOps workflow for building packages with your Azure Machine Learning workspace.
- Consider using a staging environment to test machine learning pipeline system integration with upstream or downstream application components.
- Create unit and integration tests for your inference endpoints for enhanced debugging and to accelerate time to deployment.
- To trigger retraining, use [dataset monitors](#) and [event-driven workflows](#). Subscribe to data drift events and automate the trigger of [machine learning pipelines for retraining](#).

AI factory for organization machine learning operations

A data science team might decide it can manage multiple machine learning use cases internally. Adopting machine learning operations helps an organization set up project teams for better quality, reliability, and maintainability of solutions. Through balanced teams, supported processes, and technology automation, a team that adopts machine learning operations can scale and focus on developing new use cases.

As the number of use cases grows in an organization, the management burden of supporting the use cases grows linearly, or even more. The challenge for the organization becomes how to accelerate time to market, support quicker assessment of use case feasibility, implement repeatability, and best use available resources and skill sets on a range of projects. For many organizations, developing an AI factory is the solution.

An AI factory is a system of repeatable business processes and standardized artifacts that facilitates developing and deploying a large set of machine learning use cases. An AI factory optimizes team setup, recommended practices, machine learning operations strategy, architectural patterns, and reusable templates that are tailored to business requirements.

A successful AI factory relies on repeatable processes and reusable assets to help the organization efficiently scale from tens of use cases to thousands of use cases.

The following figure summarizes key elements of an AI factory:

	Governance	Assets	MLOps	Operations	
Required	AI Factory Team setup Roles & Responsibilities Project Team setup	Reference Architectures Playbook / Document Project templates	Infrastructure as Code Model management Continuous Integration Continuous Delivery Code Repository	Logging & Monitoring Dashboards & Reports Data Drift & Retraining	
Recommended	Ethical Framework Security Cost Management	Shared libraries / packages Central hub Readiness assessments How to videos	ML Engineering team Testing approach Recommended tools Branching	Retrospectives Revisions to the assets Enablement plans by role	

Standardize on repeatable architectural patterns

Repeatability is a key characteristic of an AI factory. Data science teams can accelerate project development and improve consistency across projects by developing a few repeatable architectural patterns that cover most of the machine learning use cases for their organization. When these patterns are in place, most projects can use the patterns to get the following benefits:

- Accelerated design phase
- Accelerated approvals from IT and security teams when they reuse tools across projects
- Accelerated development due to reusable infrastructure as code templates and project templates

The architectural patterns can include but aren't limited to the following topics:

- Preferred services for each stage of the project
- Data connectivity and governance
- A machine learning operations strategy tailored to the requirements of the industry, business, or data classification
- Experiment management champion and challenger models

Facilitate cross-team collaboration and sharing

Shared code repositories and utilities can accelerate the development of machine learning solutions. Code repositories can be developed in a modular way during project development so that they're generic enough to be used in other projects. They can be made available in a central repository that all data science teams can access.

Share and reuse intellectual property

To maximize code reuse, review the following intellectual property at the beginning of a project:

- Internal code that was designed to reuse in the organization. Examples include packages and modules.
- Datasets that were created in other machine learning projects or which are available in the Azure ecosystem.
- Existing data science projects that have a similar architecture and business problems.
- GitHub or open source repositories that can accelerate the project.

Any project retrospective should include an action item to determine whether elements of the project can be shared and generalized for broader reuse. The list of assets the organization can share and reuse expands over time.

To help with sharing and discovery, many organizations have introduced shared repositories to organize code snippets and machine learning artifacts. Artifacts in Azure Machine Learning, including [datasets](#), [models](#), [environments](#), and [pipelines](#), can be defined as code, so you can share them efficiently across projects and workspaces.

Project templates

To accelerate the process of migrating existing solutions and to maximize code reuse, many organizations standardize on a project template to kickstart new projects.

Examples of project templates that are recommended for use with Azure Machine Learning are [Azure Machine Learning examples](#), the [Data Science Lifecycle Process](#), and the [Team Data Science Process](#).

Central data management

The process of getting access to data for exploration or production usage can be time consuming. Many organizations centralize data management to bring together data producers and data consumers for easier access to data for machine learning experimentation.

Shared utilities

Your organization can use enterprise-wide centralized dashboards to consolidate logging and monitoring information. The dashboards might include error logging, service availability and telemetry, and model performance monitoring.

Use Azure Monitor metrics to build a dashboard for Azure Machine Learning and associated services like Azure Storage. A dashboard helps you keep track of experimentation progress, compute infrastructure health, and GPU quota utilization.

Specialist machine learning engineering team

Many organizations have implemented the role of machine learning engineer. A machine learning engineer specializes in creating and running robust machine learning pipelines, drift monitoring and retraining workflows, and monitoring dashboards. The engineer has overall responsibility for industrializing the machine learning solution, from development to production. The engineer works closely with data engineering, architects, security, and operations to ensure that all necessary controls are in place.

Although data science requires deep domain expertise, machine learning engineering is more technical in focus. The difference makes the machine learning engineer more flexible, so they can work on various projects and with various business departments. Large data science practices might benefit from a specialist machine learning engineering team that drives repeatability and reuse of automation workflows across various use cases and business areas.

Enablement and documentation

It's important to provide clear guidance about the AI factory process for new and existing teams and users. Guidance helps ensure consistency and reduce the effort that's required from the machine learning engineering team when it industrializes a project. Consider designing content specifically for the various roles in your organization.

Everyone has a unique way of learning, so a mixture of the following types of guidance can help accelerate adoption of the AI factory framework:

- A central hub that has links to all artifacts. For example, this hub might be a channel on Microsoft Teams or a Microsoft SharePoint site.
- Training and an enablement plan designed for each role.
- A high-level summary presentation of the approach and a companion video.
- A detailed document or playbook.
- How-to videos.
- Readiness assessments.

Machine learning operations in Azure video series

A video series about [machine learning operations in Azure](#) shows you how to establish machine learning operations for your machine learning solution, from initial development to production.

Ethics

Ethics plays an instrumental role in the design of an AI solution. If ethical principles aren't implemented, trained models might exhibit the same bias that's present in the data they were trained on. The result might be that the project is discontinued. More importantly, the organization's reputation might be at risk.

To ensure that the key ethical principles that the organization stands for are implemented across projects, the organization should provide a list of these principles and ways to validate them from a technical perspective during the testing phase. Use the machine learning features in Azure Machine Learning to understand what responsible machine learning is and how to build it into your machine learning operations.

Next steps

Learn more about how to organize and set up Azure Machine Learning environments, or watch a hands-on video series about [machine learning operations in Azure](#).

[Organize and set up Azure Machine Learning environments](#)

Learn more about how to manage budgets, quotas, and costs at the organization level by using Azure Machine Learning:

[Manage machine learning budgets, costs, and quotas with Azure Machine Learning](#)

Azure Architecture Center

Design solutions on Azure using established patterns and practices. Azure Architecture Center is a catalog of solution ideas, example workloads, reference architectures, technology decision guides, and architecture guides for Azure workloads.



ARCHITECTURE
[Browse Azure architectures](#)



CONCEPT
[Build using application architecture...](#)



REFERENCE
[Make technology choices](#)



CONCEPT
[Learn cloud design patterns](#)



WHAT'S NEW
[See what's new and updated](#)

Architecting applications on Azure

Explore best practices and patterns for building applications on Microsoft Azure.



Design for the cloud

- [Learn fundamental architecture styles](#)
- [Use best practices in cloud applications](#)
- [Avoid performance anti-patterns](#)
- [Use responsible engineering practices](#)
- [Use cloud design patterns](#)



Choose the right technology

- [Choose a compute service](#)
- [Choose a container service](#)
- [Choose a data store](#)
- [Choose AI services](#)
- [Choose a messaging service](#)



Implement Azure landing zones

- [Deploy Azure landing zones](#)
- [Bicep-based landing zones](#)
- [Terraform-based landing zones](#)
- [Automate through subscription vending](#)
- [Application landing zone solutions](#)

Specialized scenarios

- [Architecture for startups](#)
- [Architecture for SaaS and multitenant applications](#)
- [Build mission-critical workloads](#)
- [Azure for AWS professionals](#)
- [Azure for Google Cloud professionals](#)

Technology areas

Explore architectures and guides for different technologies.

Popular articles

- [Baseline OpenAI end-to-end chat architecture](#)
- [Develop and optimize a RAG implementation](#)
- [Cloud design patterns](#)
- [Hub-spoke network topology](#)
- [AKS production baseline](#)
- [Choose your Azure compute service](#)
- [CQRS design pattern](#)

AI & Machine Learning

- [AI architecture design](#)
- [Azure OpenAI baseline architecture](#)
- [Build a generative AI gateway](#)
- [Develop and optimize a RAG implementation](#)
- [Use MLOps](#)
- [Use GenAIOps](#)

Analytics

- [Analytics architecture design](#)
- [Choose a data analytics technology](#)
- [Analytics end-to-end with Azure Synapse](#)
- [Enterprise business intelligence](#)
- [Near real-time lakehouse data processing](#)
- [Stream processing with Azure Databricks](#)

Compute

- [Choose an Azure compute service](#)
- [Run a Linux VM on Azure](#)
- [Run a Windows VM on Azure](#)
- [Azure VM baseline architecture](#)
- [Understand multi-region compute balancing](#)
- [Build workloads on spot virtual machines](#)

Containers

- [Choose a container service](#)
- [Design for Azure Kubernetes Service](#)
- [Azure Kubernetes Service baseline cluster](#)

Databases

- [Databases architecture design](#)
- [Migrate an Oracle database to Azure](#)
- [DataOps for a modern data warehouse](#)

- Plan Day-2 operations on AKS
- Deploy microservices with Azure Container Apps
- Deploy microservices with Azure Container Apps and Dapr

- Design a medallion lakehouse
- Use change feed to replicate data
- Intelligent apps using Azure Database for PostgreSQL

Hybrid + multicloud

- Hybrid architecture design
- Azure hybrid options
- Connect a cross-premises network to Azure
- Azure Local baseline architecture
- Hybrid Kubernetes clusters
- Hybrid file services

Identity

- Identity architecture design
- Identity in multitenant architectures
- Building applications to work across Entra tenants
- Integrate on-premises AD with Microsoft Entra ID
- Extend AD DS to Azure
- Create an AD DS forest in Azure

Networking

- Choose a hybrid network architecture
- Hub-spoke topology
- ExpressRoute with VPN failover
- Implement a secure hybrid network
- Highly available network virtual appliances
- Segment virtual networks

Security

- Security architecture design
- Azure security in AWS
- Zero-trust network with Azure Firewall and Application Gateway
- Homomorphic encryption with SEAL
- Securely managed web applications

Web apps

- Enterprise web app patterns
- Basic web application
- Baseline zone-redundant web application
- Multi-region deployment
- E-commerce product search
- Protect APIs with Application Gateway and API Management
- Serverless web application

Cloud adoption and workload design

Build a strong cloud adoption strategy and a consistent approach to workload design.



[Cloud Adoption Framework for Azure](#)



[Azure Well-Architected Framework pillars](#)



[Well-architected workloads](#)



[Well-architected service guides](#)

Cloud Operating Model is now part of the Microsoft Cloud Adoption Framework for Azure

Article • 02/28/2023

In early 2018, Microsoft released the Cloud Operating Model (COM). The COM was a guide that helped customers understand the *what* and the *why* of digital transformation. This helped customers get a sense of all the areas that needed to be addressed: business strategy, culture strategy, and technology strategy. What was not included in the COM were the specific *how-to* steps, which left customers wondering, "Where do we go from here?"

The Microsoft Cloud Adoption Framework for Azure, is designed to help you understand the **what** and **why** and provide unified guidance on the **how** to help accelerate your cloud adoption efforts.

Using Cloud Operating Model practices within the Cloud Adoption Framework

For an approach that's similar to the COM, begin with one of the following:

- [Get started: Accelerate migration](#)
- [Get started: Build and innovate in the cloud](#)
- [Enable success with a sound operating model](#)

The virtual datacenter: A network perspective

Article • 02/28/2023

Applications migrated from on-premises might benefit from Azure's secure cost-efficient infrastructure, even with minimal application changes. Enterprises might want to adapt their architectures to improve agility and take advantage of Azure's capabilities.

Microsoft Azure delivers hyperscale services and infrastructure with enterprise-grade capabilities and reliability. These services and infrastructure offer many choices in hybrid connectivity, which allows customers to access them over the internet or a private network connection. Microsoft partners can also provide enhanced capabilities by offering security services and virtual appliances that are optimized to run in Azure.

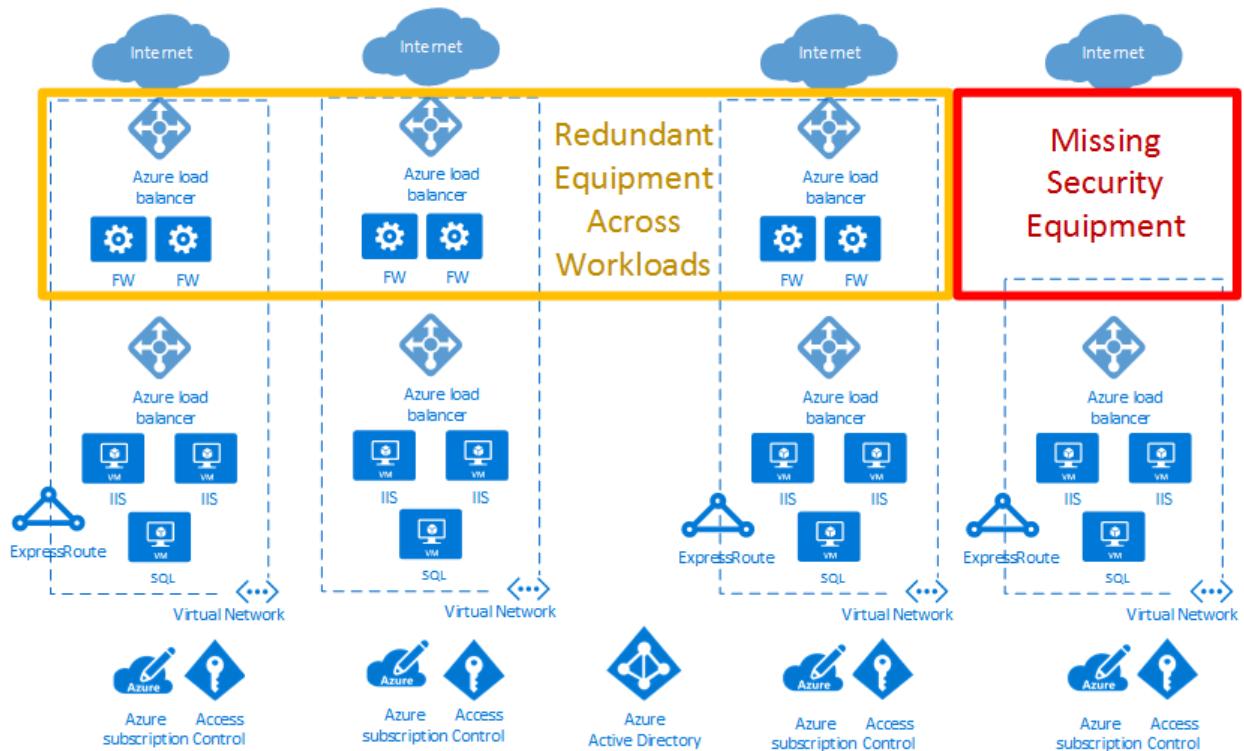
Customers can use Azure to seamlessly extend their infrastructure into the cloud and build multitier architectures.

What is a virtual datacenter?

The cloud began as a platform for hosting public-facing applications. Enterprises recognized the value of the cloud and began migrating internal line-of-business applications. These applications brought more security, reliability, performance, and cost considerations that required more flexibility when delivering cloud services. New infrastructure and networking services were designed to provide flexibility. New features provide elastic scale, disaster recovery, and other considerations.

Cloud solutions were initially designed to host single, relatively isolated applications in the public spectrum, which worked well for a few years. As the benefits of cloud solutions became clear, multiple large-scale workloads were hosted on the cloud. Addressing security, reliability, performance, and cost concerns is vital for the deployment and lifecycle of your cloud service.

In the example cloud deployment diagram below, the red box highlights a security gap. The yellow box shows an opportunity to optimize network virtual appliances across workloads.



Virtual datacenters help achieve the scale required for enterprise workloads. The scale must address the challenges introduced when running large-scale applications in the public cloud.

A virtual datacenter implementation includes more than the application workloads in the cloud. It also provides network, security, management, DNS, and Active Directory services. As enterprises migrate more workloads to Azure, consider the infrastructure and objects that support these workloads. Good resource management helps avoid the increase of separately managed "workload islands" with independent data flows, security models, and compliance challenges.

The virtual datacenter concept provides recommendations and high-level designs for implementing a collection of separate but related entities. These entities often have common supporting functions, features, and infrastructure. Viewing your workloads as a virtual datacenter helps realize reduced cost from economies of scale. It also helps with optimized security via component and data flow centralization, and easier operations, management, and compliance audits.

ⓘ Note

A virtual datacenter isn't a specific Azure service. Rather, various Azure features and capabilities are combined to meet your requirements. A virtual datacenter is a way of thinking about your workloads and Azure usage to optimize your resources and capabilities in the cloud. It provides a modular approach to providing IT services in Azure, while respecting the enterprise's organizational roles and responsibilities.

A virtual datacenter helps enterprises deploy workloads and applications in Azure for the following scenarios:

- Host multiple related workloads.
- Migrate workloads from an on-premises environment to Azure.
- Implement shared or centralized security and access requirements across workloads.
- Mix DevOps and centralized IT appropriately for a large enterprise.

Who should implement a virtual datacenter?

Any customer who decides to adopt Azure can benefit from the efficiency of configuring a set of resources for common use by all applications. Depending on the size, even single applications can benefit from using the patterns and components used to build a VDC implementation.

Some organizations have centralized teams or departments for IT, networking, security, or compliance. Implementing a VDC can help enforce policy points, separate responsibilities, and ensure the consistency of underlying common components.

Application teams can retain the freedom and control that is suitable for their requirements.

Organizations with a DevOps approach can also use VDC concepts to provide authorized pockets of Azure resources. This method ensures the DevOps groups have total control within that grouping, at either the subscription level or within resource groups in a common subscription. At the same time, network and security boundaries stay compliant. Compliance is defined by a centralized policy in the hub network and centrally managed resource group.

Considerations for implementing a virtual datacenter

When designing a virtual datacenter, consider these pivotal issues:

Identity and directory service

Identity and directory services are key capabilities of both on-premises and cloud datacenters. Identity covers all aspects of access and authorization to services within a VDC implementation. To ensure that only authorized users and processes access your Azure resources, Azure uses several types of credentials for authentication, including

account passwords, cryptographic keys, digital signatures, and certificates. [Microsoft Entra multifactor authentication](#) provides an extra layer of security for accessing Azure services. A strong authentication with a range of easy verification options (phone call, text message, or mobile app notification) allows customers to choose the method they prefer.

Large enterprises need to define identity management processes that describe the management of individual identities, their authentication, authorization, roles, and privileges within or across their VDC. The goals of this process might increase security and productivity, while reducing cost, downtime, and repetitive manual tasks.

Enterprise organizations might require a demanding mix of services for different lines of business. Employees often have different roles when involved with different projects. The VDC requires good cooperation between different teams, each with specific role definitions to get systems running with good governance. The matrix of responsibilities, access, and rights can be complex. Identity management in the VDC is implemented through [Microsoft Entra ID](#) and Azure role-based access control (Azure RBAC).

A directory service is a shared information infrastructure that locates, manages, administers, and organizes everyday items and network resources. These resources can include volumes, folders, files, printers, users, groups, devices, and other objects. Each resource on the network is considered an object by the directory server. Information about a resource is stored as a collection of attributes associated with that resource or object.

All Microsoft online business services rely on Microsoft Entra ID for sign-on and other identity needs. Microsoft Entra ID is a comprehensive, highly available identity and access management cloud solution that combines core directory services, advanced identity governance, and application access management. Microsoft Entra ID can integrate with on-premises Active Directory to enable single sign-on for all cloud-based and locally hosted on-premises applications. The user attributes of on-premises Active Directory can be automatically synchronized to Microsoft Entra ID.

Each specific department, group of users, or services in the Directory Service must have the minimum permissions required to manage their own resources within a VDC implementation. Structuring permissions requires balancing. Too many permissions can impede performance efficiency, and too few or loose permissions can increase security risks. Azure role-based access control (Azure RBAC) helps to address this problem by offering fine-grained access management for resources in a VDC implementation.

Security infrastructure

Security infrastructure refers to the segregation of traffic in a VDC implementation's specific virtual network segment. This infrastructure specifies how ingress and egress are controlled in a VDC implementation. Azure is based on a multitenant architecture that prevents unauthorized and unintentional traffic between deployments. This is done by using virtual network isolation, access control lists, load balancers, IP filters, and traffic flow policies. Network address translation (NAT) separates internal network traffic from external traffic.

The Azure fabric allocates infrastructure resources to tenant workloads and manages communications to and from Virtual Machines (VMs). The Azure hypervisor enforces memory and process separation between VMs and securely routes network traffic to guest OS tenants.

Connectivity to the cloud

A virtual datacenter requires connectivity to external networks to offer services to customers, partners, or internal users. This need for connectivity refers not only to the Internet, but also to on-premises networks and datacenters.

Customers control the services that can access and be accessed from the public internet. This access is controlled by using [Azure Firewall](#) or other types of virtual network appliances (NVAs), custom routing policies by using [user-defined routes](#), and network filtering by using [network security groups](#). We recommend that all internet-facing resources are protected by the [Azure DDoS Protection](#).

Enterprises might need to connect their virtual datacenter to on-premises datacenters or other resources. This connectivity between Azure and on-premises networks is a crucial aspect when designing an effective architecture. Enterprises have two different ways to create this interconnection: transit over the Internet or via private direct connections.

An [Azure Site-to-Site VPN](#) connects on-premises networks to your virtual datacenter in Azure. The link is established through secure encrypted connections (IPsec tunnels). Azure Site-to-Site VPN connections are flexible, quick to create, and typically don't require any more hardware procurement. Based on industry standard protocols, most current network devices can create VPN connections to Azure over the internet or existing connectivity paths.

[ExpressRoute](#) enables private connections between your virtual datacenter and any on-premises networks. ExpressRoute connections don't go over the public Internet, and offer higher security, reliability, and higher speeds (up to 100 Gbps) along with consistent latency. ExpressRoute provides the benefits of compliance rules associated

with private connections. With [ExpressRoute Direct](#), you can connect directly to Microsoft routers at either 10 Gbps or 100 Gbps.

Deploying ExpressRoute connections usually involves engaging with an ExpressRoute service provider (ExpressRoute Direct being the exception). For customers that need to start quickly, it's common to initially use Site-to-Site VPN to establish connectivity between a virtual datacenter and on-premises resources. Once your physical interconnection with your service provider is complete, migrate connectivity over your ExpressRoute connection.

For large numbers of VPN or ExpressRoute connections, [Azure Virtual WAN](#) is a networking service that provides optimized and automated branch-to-branch connectivity through Azure. Virtual WAN lets you connect to and configure branch devices to communicate with Azure. Connecting and configuring can be done either manually or by using preferred provider devices through a Virtual WAN partner. Using preferred provider devices allows ease of use, simplification of connectivity, and configuration management. The Azure WAN built-in dashboard provides instant troubleshooting insights that can help save you time, and gives you an easy way to view large-scale site-to-site connectivity. Virtual WAN also provides security services with an optional Azure Firewall and Firewall Manager in your Virtual WAN hub.

Connectivity within the cloud

[Azure Virtual Networks](#) and [virtual network peering](#) are the basic networking components in a virtual datacenter. A virtual network guarantees an isolation boundary for virtual datacenter resources. Peering allows intercommunication between different virtual networks within the same Azure region, across regions, and even between networks in different subscriptions. Traffic flows can be controlled inside and between virtual networks by sets of security rules specified for [network security groups](#), firewall policies ([Azure Firewall](#) or [network virtual appliances](#)), and custom [user-defined routes](#).

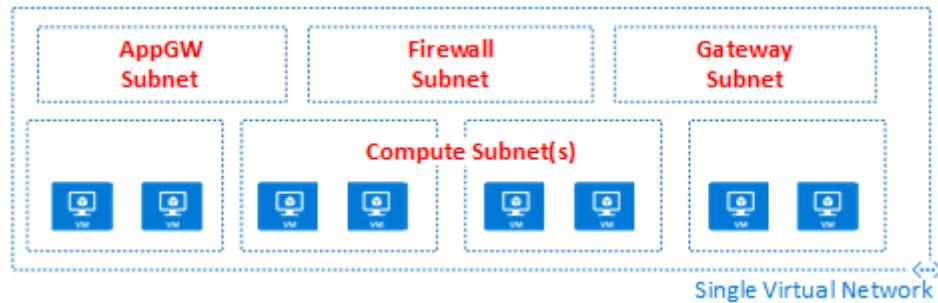
Virtual networks are anchor points for integrating platform as a service (PaaS) Azure products like [Azure Storage](#), [Azure SQL](#), and other integrated public services that have public endpoints. With [service endpoints](#) and [Azure Private Link](#), you can integrate your public services with your private network. You can even take your public services private, but still enjoy the benefits of Azure-managed PaaS services.

Virtual datacenter overview

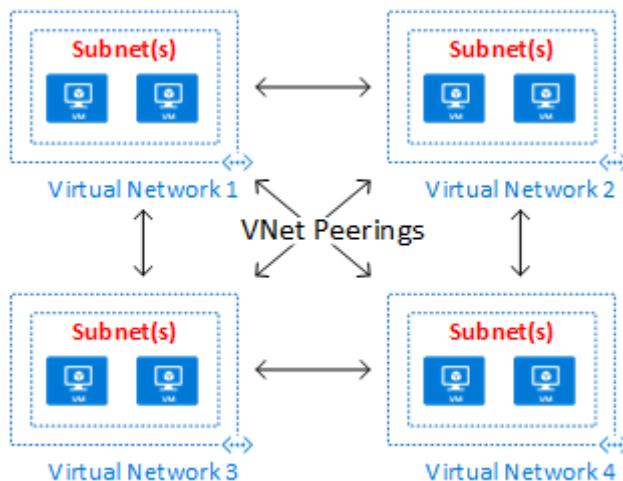
Topologies

A virtual datacenter can be built using one of these high-level topologies, based on your needs and scale requirements:

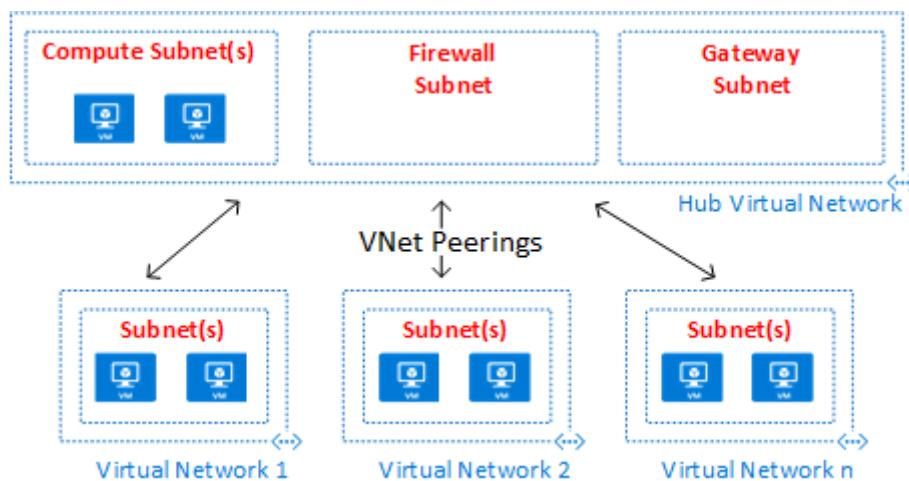
In a *Flat topology*, all resources are deployed in a single virtual network. Subnets allow for flow control and segregation.



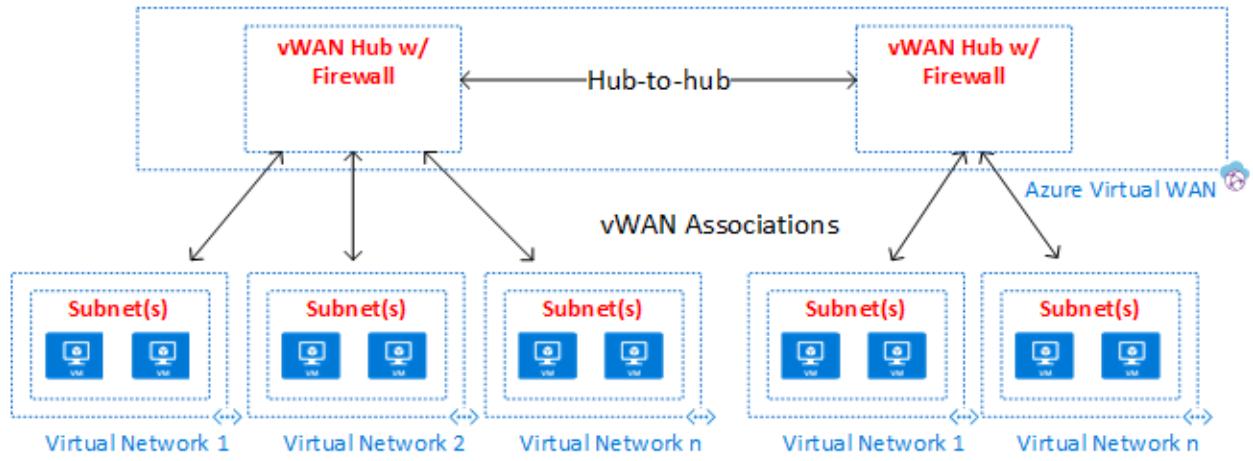
In a *Mesh topology*, virtual network peering connects all virtual networks directly to each other.



A *Peering hub and spoke topology* is well suited for distributed applications and teams with delegated responsibilities.



An *Azure Virtual WAN topology* can support large-scale branch office scenarios and global WAN services.



The peering hub and spoke topology and the Azure Virtual WAN topology both use a hub and spoke design, which is optimal for communication, shared resources, and centralized security policy. Hubs are built using either a virtual network peering hub (labeled as `Hub Virtual Network` in the diagram) or a Virtual WAN hub (labeled as `Azure Virtual WAN` in the diagram). Azure Virtual WAN is designed for large-scale branch-to-branch and branch-to-Azure communications, or for avoiding the complexities of building all the components individually in a virtual networking peering hub. In some cases, your requirements might mandate a virtual network peering hub design, such as the need for network virtual appliances in the hub.

In hub and spoke topologies, the hub is the central network zone that controls and inspects all traffic between different zones such as the internet, on-premises, and the spokes. The hub and spoke topology helps the IT department centrally enforce security policies. It also reduces the potential for misconfiguration and exposure.

The hub often contains common service components consumed by the spokes. The following examples are common central services:

- The Windows Active Directory infrastructure is required for user authentication of third parties that access from untrusted networks before they get access to the workloads in the spoke. It includes the related Active Directory Federation Services (AD FS)
- A Distributed Name System (DNS) service is used to resolve naming for the workload in the spokes and to access resources on-premises and on the internet if `Azure DNS` isn't used
- A public key infrastructure (PKI) is used to implement single sign-on on workloads
- Flow control of TCP and UDP traffic between the spoke network zones and the internet
- Flow control between the spokes and on-premises
- If needed, flow control between one spoke and another

A virtual datacenter reduces overall cost by using the shared hub infrastructure between multiple spokes.

The role of each spoke can be to host different types of workloads. The spokes also provide a modular approach for repeatable deployments of the same workloads. Examples include dev/test, user acceptance testing, preproduction, and production. The spokes can also segregate and enable different groups within your organization. DevOps groups are a good example of what spokes can do. Inside a spoke, it's possible to deploy a basic workload or complex multitier workloads with traffic control between the tiers.

Subscription limits and multiple hubs

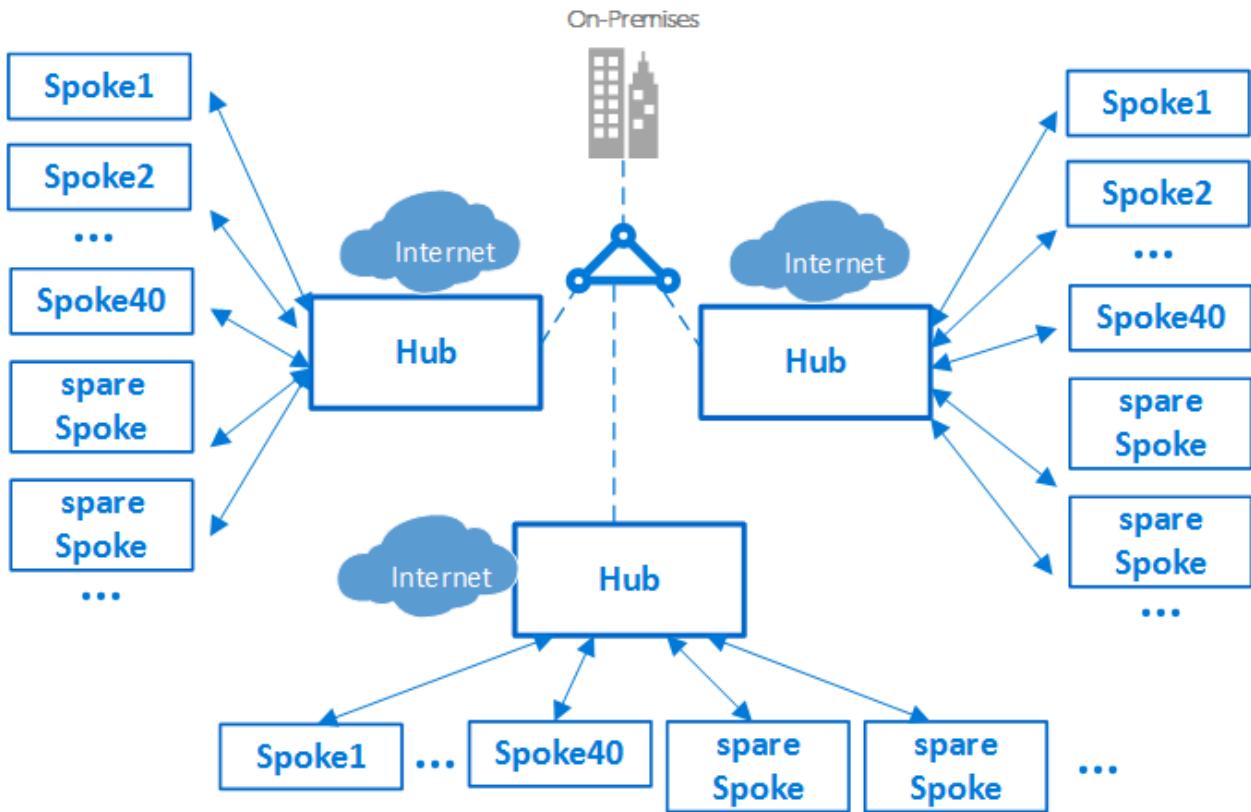
Important

Based on the size of your Azure deployments, you might need a multiple hub strategy. When designing your hub and spoke strategy, ask "Can this design scale to use another hub virtual network in this region?" and "Can this design scale accommodate multiple regions?" It's far better to plan for a design that scales and not need it, than to fail to plan and need it.

When to scale to a secondary (or more) hub depends on several factors, usually based on inherent limits on scale. Be sure to review the subscription, virtual network, and virtual machine [limits](#) when designing for scale.

In Azure, every component, whatever the type, is deployed in an Azure subscription. The isolation of Azure components in different Azure subscriptions can satisfy the requirements of different lines of business, such as setting up differentiated levels of access and authorization.

A single VDC implementation can scale up a large number of spokes. Although, as with every IT system, there are platform limits. The hub deployment is bound to a specific Azure subscription, which has restrictions and limits (for example, a maximum number of virtual network peerings. For details, see [Azure subscription and service limits, quotas, and constraints](#)). In cases where limits might be an issue, the architecture can scale up further by extending the model from a single hub-spokes to a cluster of hub and spokes. Multiple hubs in one or more Azure regions can be connected using virtual network peering, ExpressRoute, Virtual WAN, or Site-to-Site VPN.

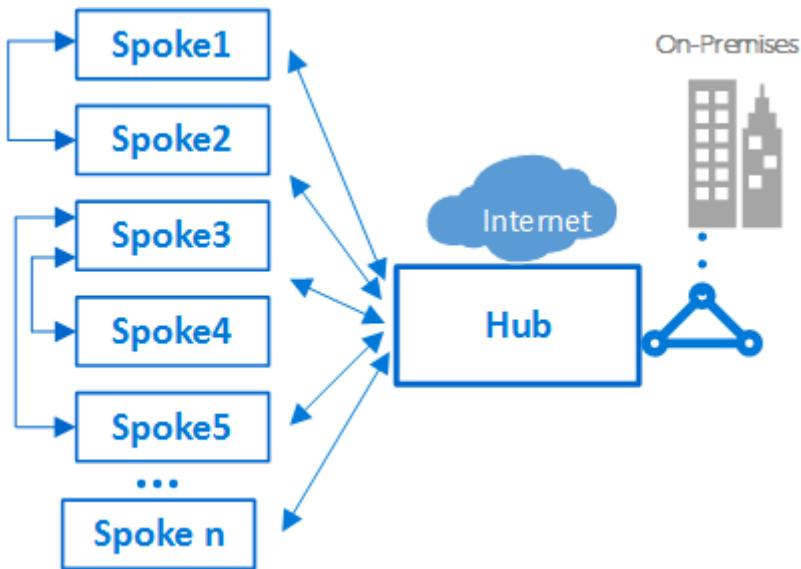


The introduction of multiple hubs increases the cost and management effort of the system. It's only justified due to scalability, system limits, redundancy, regional replication for end-user performance, or disaster recovery. In scenarios requiring multiple hubs, all the hubs should strive to offer the same set of services for operational ease.

Interconnection between spokes

Inside a single spoke, or a flat network design, it's possible to implement complex multitier workloads. Multitier configurations can be implemented using subnets, which are one for every tier or application in the same virtual network. Traffic control and filtering are done using network security groups and user-defined routes.

An architect might want to deploy a multitier workload across multiple virtual networks. With virtual network peering, spokes can connect to other spokes in the same hub or different hubs. A typical example of this scenario is the case where application processing servers are in one spoke, or virtual network. The database deploys in a different spoke, or virtual network. In this case, it's easy to interconnect the spokes with virtual network peering, which avoids transiting through the hub. Complete a careful architecture and security review to ensure that bypassing the hub doesn't bypass important security or auditing points that might exist only in the hub.



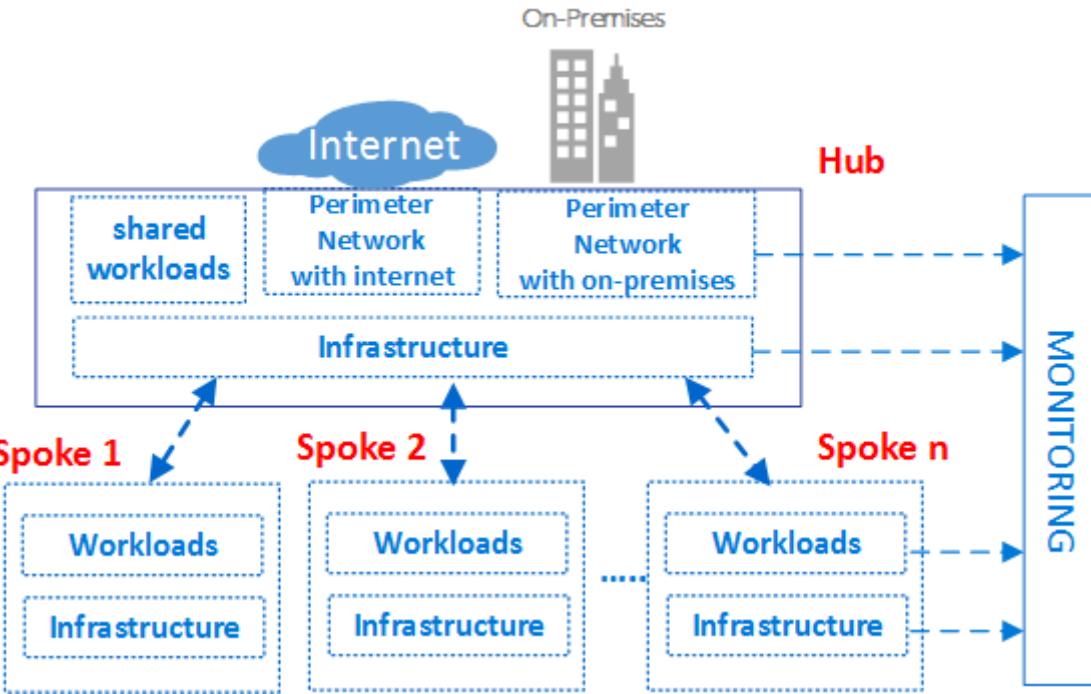
Spokes can also interconnect to a spoke that acts as a hub. This approach creates a two-level hierarchy. The spoke in the higher level (level 0) becomes the hub of lower spokes (level 1) of the hierarchy. The spokes for a VDC implementation are required to forward the traffic to the central hub. The traffic can then transit to its destination in either the on-premises network or the public internet. An architecture with two levels of hubs introduces complex routing that removes the benefits of a simple hub-spoke relationship.

Although Azure allows complex topologies, one of the core principles of the VDC concept is repeatability and simplicity. To minimize management effort, the simple hub-spoke design is the VDC reference architecture that we recommend.

Components

The virtual datacenter is made up of four basic component types: **Infrastructure**, **Perimeter Networks**, **Workloads**, and **Monitoring**.

Each component type consists of various Azure features and resources. Your VDC implementation is made up of instances of multiple component types and multiple variations of the same component type. For instance, you might have many different, logically separated workload instances that represent different applications. You use these different component types and instances to build the VDC.



The preceding high-level conceptual architecture of the VDC shows different component types used in different zones of the hub-spokes topology. The diagram shows infrastructure components in various parts of the architecture.

As good practice in general, access rights and privileges can be group-based. Dealing with groups rather than individual users eases maintenance of access policies, by providing a consistent way to manage it across teams, which aids in minimizing configuration errors. Assigning and removing users to and from appropriate groups helps keep the privileges of a specific user up to date.

Each role group can have a unique prefix on their names. This prefix makes it easy to identify which workload a group is associated with. For example, a workload hosting an authentication service might have groups named **AuthServiceNetOps**, **AuthServiceSecOps**, **AuthServiceDevOps**, and **AuthServiceInfraOps**. Centralized roles, or roles not related to a specific service, might be prefaced with **Corp**. An example is **CorpNetOps**.

Many organizations use a variation of the following groups to provide a major breakdown of roles:

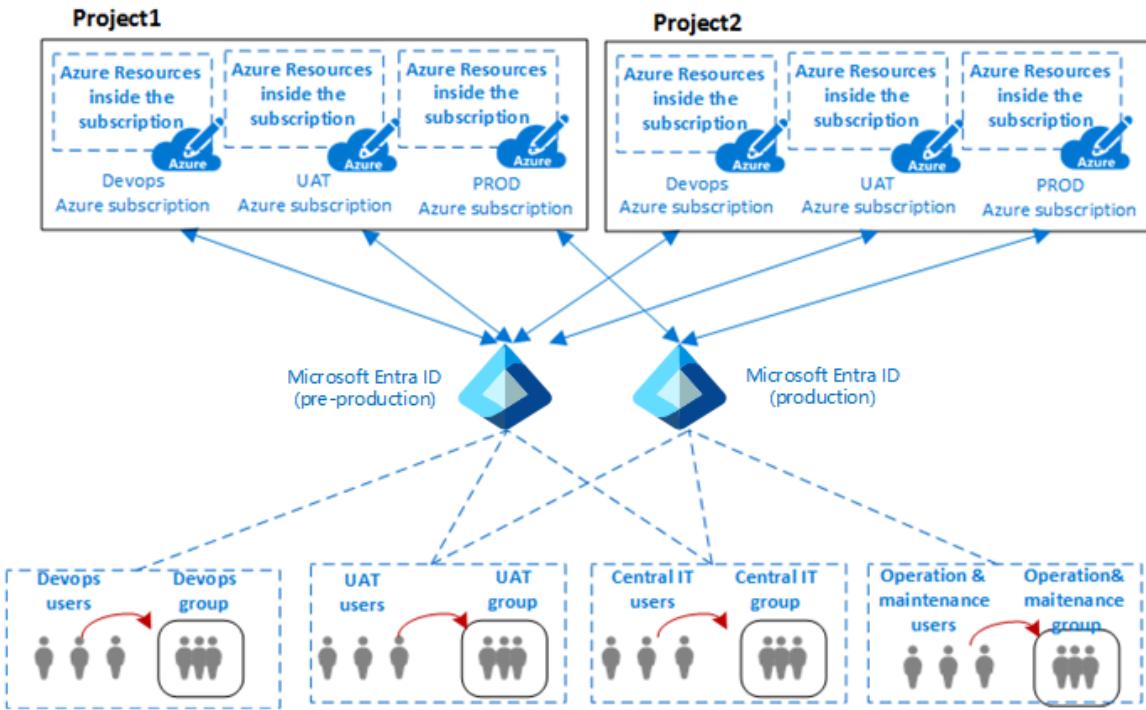
- The central IT team named **Corp** has the ownership rights to control infrastructure components. Examples are networking and security. The group needs to have the role of contributor on the subscription, control of the hub, and network contributor rights in the spokes. Large organizations frequently split up these management responsibilities between multiple teams. Examples are a network operations **CorpNetOps** group with exclusive focus on networking and a security operations **CorpSecOps** group responsible for the firewall and security policy. In

this specific case, two different groups need to be created for assignment of these custom roles.

- The dev/test group named **AppDevOps** has the responsibility to deploy app or service workloads. This group takes the role of virtual machine contributor for IaaS deployments or one or more PaaS contributor's roles. For more information, see [Azure built-in roles](#). Optionally, the dev/test team might need visibility on security policies (network security groups) and routing policies (user-defined routes) inside the hub or a specific spoke. In addition to the role of contributor for workloads, this group would also need the role of network reader.
- The operation and maintenance group called **CorpInfraOps** or **AppInfraOps** has the responsibility of managing workloads in production. This group needs to be a subscription contributor on workloads in any production subscriptions. Some organizations might also evaluate if they need an escalation support team group with the role of subscription contributor in production and the central hub subscription. The other group fixes potential configuration issues in the production environment.

The VDC is designed so that central IT team groups that manage the hub have corresponding groups at the workload level. In addition to managing hub resources, the central IT team can control external access and top-level permissions on the subscription. Workload groups can also control resources and permissions of their virtual network independently from the central IT team.

The virtual datacenter is partitioned to securely host multiple projects across different lines of business. All projects require different isolated environments (dev, UAT, and production). Separate Azure subscriptions for each of these environments can provide natural isolation.



The preceding diagram shows the relationship between an organization's projects, users, groups, and the environments where the Azure components are deployed.

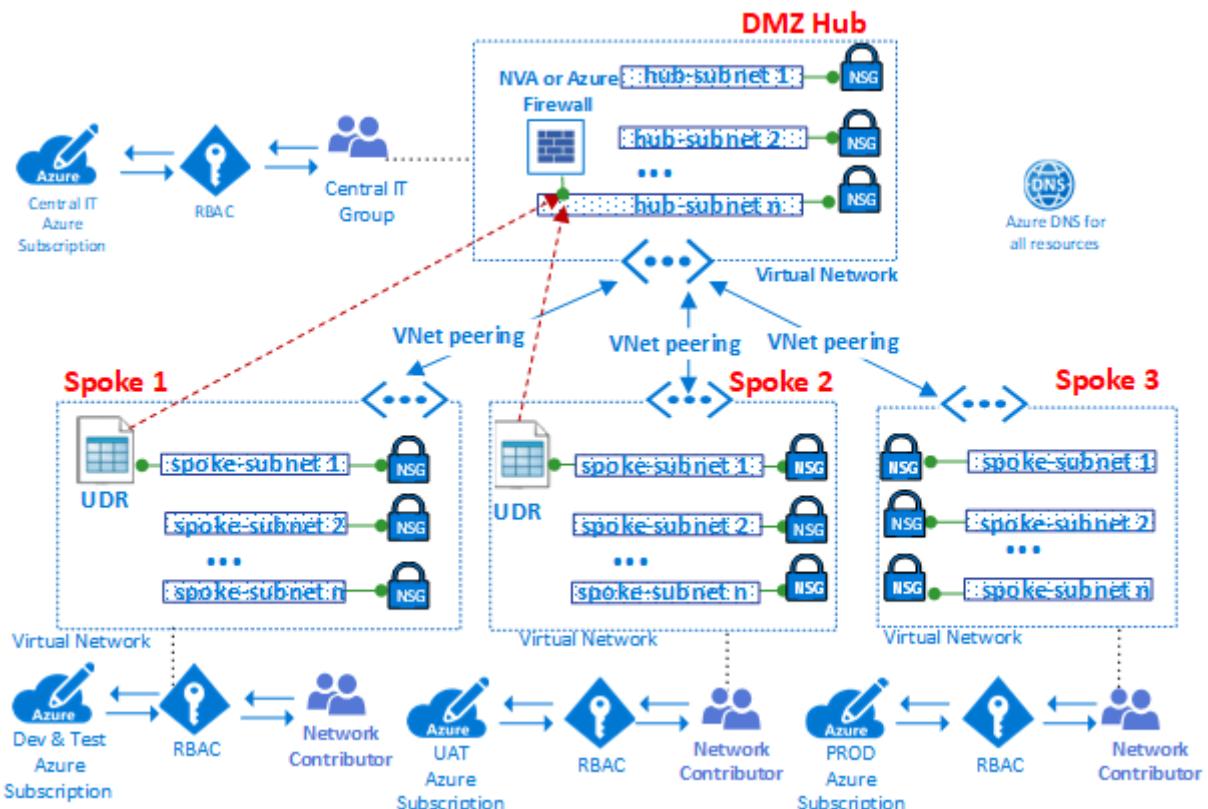
Typically in IT, an environment (or tier) is a system in which multiple applications are deployed and executed. Large enterprises use a development environment (where changes are made and tested) and a production environment (what end-users use). Those environments are separated, often with several staging environments in between them, to allow phased deployment (rollout), testing, and rollback if problems arise. Deployment architectures vary significantly, but usually the basic process of starting at development (DEV) and ending at production (PROD) is still followed.

A common architecture for these types of multitier environments includes DevOps for development and testing, UAT for staging, and production environments. Organizations can use single or multiple Microsoft Entra tenants to define access and rights to these environments. The previous diagram shows a case where two different Microsoft Entra tenants are used: one for DevOps and UAT, and the other exclusively for production.

The presence of different Microsoft Entra tenants enforces the separation between environments. The same group of users, such as the central IT team, needs to authenticate by using a different URI to access a different Microsoft Entra tenant. This allows the team to modify the roles or permissions of either the DevOps or production environments of a project. The presence of different user authentications to access different environments reduces possible outages and other issues caused by human errors.

Component type: infrastructure

This component type is where most of the supporting infrastructure resides. It's also where your centralized IT, security, and compliance teams spend most of their time.



Infrastructure components provide an interconnection for the different components of a VDC implementation, and are present in both the hub and the spokes. The responsibility for managing and maintaining the infrastructure components is typically assigned to the central IT team or security team.

One of the primary tasks of the IT infrastructure team is to guarantee the consistency of IP address schemas across the enterprise. The private IP address space assigned to a VDC implementation must be consistent and not overlapping with private IP addresses assigned on your on-premises networks.

While NAT on the on-premises edge routers or in Azure environments can avoid IP address conflicts, it adds complications to your infrastructure components. Simplicity of management is one of the key goals of the VDC. Using NAT to handle IP concerns, while a valid solution, isn't a recommended solution.

Infrastructure components have the following functionality:

- **Identity and directory services:** Access to every resource type in Azure is controlled by an identity stored in a directory service. The directory service stores not only the list of users, but also the access rights to resources in a specific Azure subscription.

These services can exist in the cloud, or they can be synchronized with on-premises identity that's stored in Active Directory.

- **Virtual networks:** Virtual networks are one of main components of the VDC, and enable you to create a traffic isolation boundary on the Azure platform. A virtual network is composed of a single or multiple virtual network segments, each with a specific IP network prefix (a subnet, either IPv4 or dual stack IPv4/IPv6). The virtual network defines an internal perimeter area where IaaS virtual machines and PaaS services can establish private communications. VMs (and PaaS services) in one virtual network can't communicate directly to VMs (and PaaS services) in a different virtual network. This is true even if both virtual networks are created by the same customer, under the same subscription. Isolation is a critical property that ensures customer VMs and communication remains private within a virtual network. Where cross-network connectivity is desired, the following features describe how it can be accomplished.
- **Virtual network peering:** The fundamental feature used to create the infrastructure of a VDC is virtual network peering, which connects two virtual networks in the same region. This connection happens through the Azure datacenter network or using the Azure worldwide backbone across regions.
- **Virtual Network service endpoints:** Service endpoints extend your virtual network private address space to include your PaaS space. The endpoints also extend the identity of your virtual network to the Azure services over a direct connection. Endpoints allow you to secure your critical Azure service resources to your virtual networks.
- **Private Link:** Azure Private Link enables you to access Azure PaaS Services (for example, [Azure Storage](#), [Azure Cosmos DB](#), and [Azure SQL Database](#)) and Azure hosted customer/partner services over a Private Endpoint in your virtual network. Traffic between your virtual network and the service traverses over the Microsoft backbone network, eliminating exposure from the public Internet. You can also create your own [Private Link Service](#) in your virtual network and deliver it privately to your customers. The setup and consumption experience using Azure Private Link is consistent across Azure PaaS, customer-owned, and shared partner services.
- **User-defined routes:** Traffic in a virtual network is routed by default based on the system routing table. A user-defined route is a custom routing table that network administrators can associate to one or more subnets to override the behavior of the system routing table and defines a communication path within a virtual network. The presence of user-defined routes guarantees traffic from the spoke transit through specific custom VMs or network virtual appliances and load balancers present in both the hub and the spokes.
- **Network security groups:** A network security group is a list of security rules that act as traffic filtering on IP sources, IP destinations, protocols, IP source ports, and IP destination ports (also called a Layer 4 five-tuple). The network security group can

be applied to a subnet, a Virtual NIC associated with an Azure VM, or both. The network security groups are essential to implement a correct flow control in the hub and in the spokes. The level of security afforded by the network security group is a function of which ports you open, and for what purpose. Customers can apply more per-VM filters with host-based firewalls such as iptables or Windows Firewall.

- **DNS:** DNS provides name resolution for resources in a virtual datacenter. Azure provides DNS services for both [public](#) and [private](#) name resolution. Private zones provide name resolution within a virtual network and across virtual networks. Private zones can span across virtual networks in the same region, and across regions and subscriptions. For public resolution, Azure DNS provides a hosting service for DNS domains, providing name resolution using Microsoft Azure infrastructure. By hosting your domains in Azure, you can manage your DNS records using the same credentials, APIs, tools, and billing as your other Azure services.
- **Management group, subscription, and resource group management.** A subscription defines a natural boundary to create multiple groups of resources in Azure. This separation can be for function, role segregation, or billing. Resources in a subscription are assembled together in logical containers known as resource groups. The resource group represents a logical group to organize the resources in a virtual datacenter. If your organization has many subscriptions, you might need a way to efficiently manage access, policies, and compliance for those subscriptions. Azure management groups provide a level of scope above subscriptions. You organize subscriptions into containers known as management groups and apply your governance conditions to the management groups. All subscriptions within a management group automatically inherit the conditions applied to the management group. To see these three features in a hierarchy view, see [Organizing your resources](#) in the Cloud Adoption Framework.
- **Azure role-based access control (Azure RBAC):** Azure RBAC can map organizational roles and rights to access specific Azure resources. This allows you to restrict users to only a certain subset of actions. If you're synchronizing Microsoft Entra ID with an on-premises Active Directory, you can use the same Active Directory groups in Azure that you use on-premises. With Azure RBAC, you can grant access by assigning the appropriate role to users, groups, and applications within the relevant scope. The scope of a role assignment can be an Azure subscription, a resource group, or a single resource. Azure RBAC allows inheritance of permissions. A role assigned at a parent scope also grants access to the children contained within it. With Azure RBAC, you can segregate duties, and grant only the amount of access to users they need to perform their jobs. For example, one employee can manage virtual machines in a subscription, while another can manage SQL Server databases in the same subscription.

Component Type: Perimeter Networks

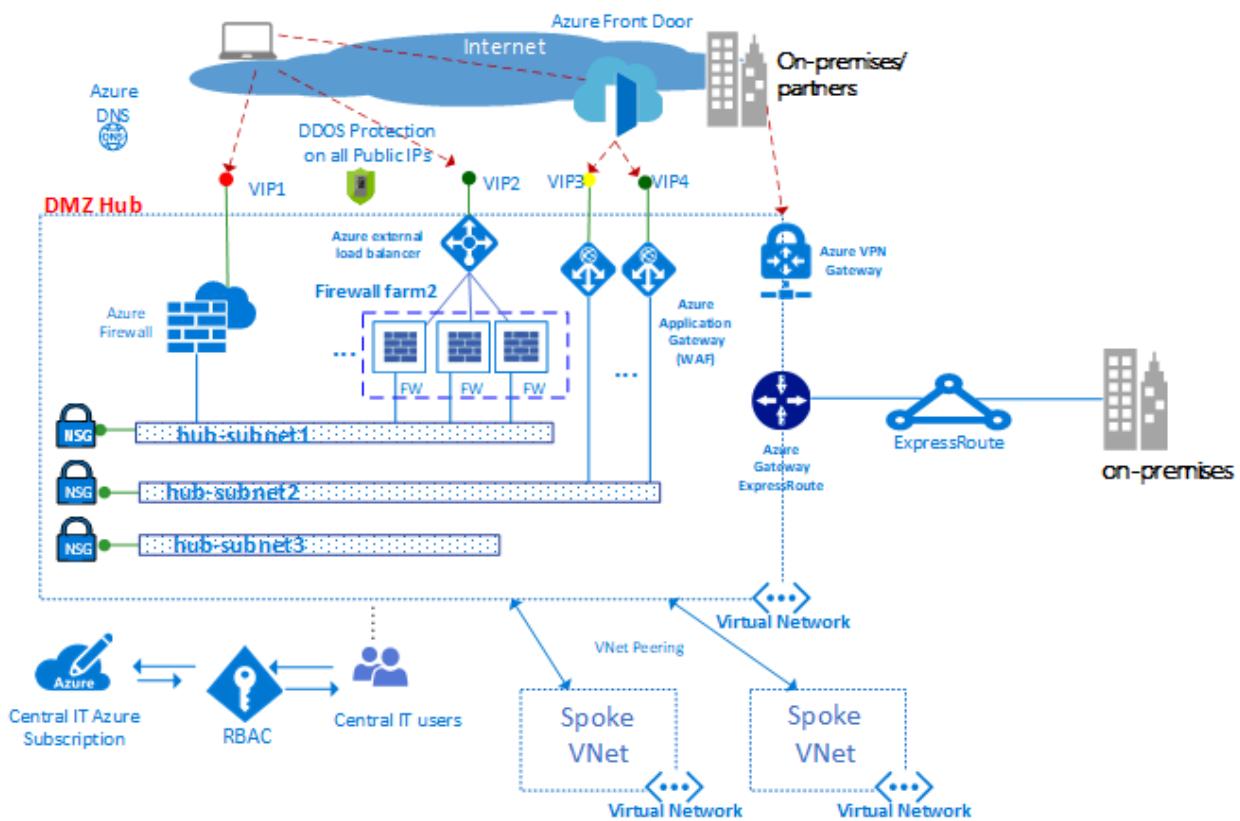
Components of a perimeter network (sometimes called a DMZ network) connect your on-premises or physical datacenter networks, along with any internet connectivity. The perimeter typically requires a significant time investment from your network and security teams.

Incoming packets can flow through the security appliances in the hub before reaching the back-end servers and services in the spokes. Examples include the firewall, IDS, and IPS. Before they leave the network, internet-bound packets from the workloads can also flow through the security appliances in the perimeter network. This flow enables policy enforcement, inspection, and auditing.

Perimeter network components include:

- [Virtual networks](#), [user-defined routes](#), and [network security groups](#)
- [Network virtual appliances](#)
- [Azure Load Balancer](#)
- [Azure Application Gateway](#) with [web application firewall \(WAF\)](#)
- [Public IPs](#)
- [Azure Front Door](#) with [web application firewall \(WAF\)](#)
- [Azure Firewall](#) and [Azure Firewall Manager](#)
- [Standard DDoS Protection](#)

Usually, the central IT team and security teams have responsibility for requirement definition and operation of the perimeter networks.



The preceding diagram shows the enforcement of two perimeters with access to the internet and an on-premises network, both resident in the DMZ hub. In the DMZ hub, the perimeter network to internet can scale up to support many lines of business, using multiple farms of Web Application Firewalls (WAFs) or Azure Firewalls. The hub also allows for on-premises connectivity via VPN or ExpressRoute as needed.

ⓘ Note

In the preceding diagram, in the **DMZ Hub**, many of the following features can be bundled together in an Azure Virtual WAN hub (such as virtual networks, user-defined routes, network security groups, VPN gateways, ExpressRoute gateways, Azure Load Balancers, Azure Firewalls, Firewall Manager, and DDOS). Using Azure Virtual WAN hubs can make the creation of the hub virtual network and the VDC much easier, since most of the engineering complexity is handled for you by Azure when you deploy an Azure Virtual WAN hub.

Virtual networks. The hub is typically built on a virtual network with multiple subnets that host different types of services. These services filter and inspect traffic to or from the internet via Azure Firewall, NVAs, WAF, and Azure Application Gateway instances.

User-defined routes. By using user-defined routes, customers can deploy firewalls, IDS/IPS, and other virtual appliances. They can route network traffic through these security appliances for security boundary policy enforcement, auditing, and inspection. User-defined routes can be created in both the hub and the spokes to guarantee that

traffic transits through the specific custom VMs, Network Virtual Appliances, and load balancers used by a VDC implementation. To guarantee that traffic generated from virtual machines in the spoke transits to the correct virtual appliances, a user-defined route needs to be set in the subnets of the spoke. This is done by setting the front-end IP address of the internal load balancer as the next hop. The internal load balancer distributes the internal traffic to the virtual appliances (load balancer back-end pool).

[Azure Firewall](#) is a managed network security service that protects your Azure Virtual Network resources. It's a stateful managed firewall with high availability and cloud scalability. You can centrally create, enforce, and log application and network connectivity policies across subscriptions and virtual networks. Azure Firewall uses a static public IP address for your virtual network resources. It allows outside firewalls to identify traffic that originates from your virtual network. The service is fully integrated with Azure Monitor for logging and analytics.

If you use the Azure Virtual WAN topology, the [Azure Firewall Manager](#) is a security management service that provides central security policy and route management for cloud-based security perimeters. It works with Azure Virtual WAN hub, a Microsoft-managed resource that lets you easily create hub and spoke architectures. When security and routing policies are associated with a hub, it's referred to as a secured virtual hub.

[Network virtual appliances](#). In the hub, the perimeter network with access to the internet is normally managed through an Azure Firewall instance or a farm of firewalls or web application firewall (WAF).

Different lines of business commonly use many web applications, which tend to suffer from various vulnerabilities and potential exploits. Web application firewalls are a special type of product used to detect attacks against web applications and HTTP/HTTPS more effectively than a generic firewall. Compared with tradition firewall technology, WAFs have a set of specific features to protect internal web servers from threats.

An Azure Firewall or NVA firewall use a common administration plane, with a set of security rules to protect the workloads hosted in the spokes, and control access to on-premises networks. The Azure Firewall has scalability built in, whereas NVA firewalls can be manually scaled behind a load balancer. Generally, a firewall farm has less specialized software compared with a WAF, but has a broader application scope to filter and inspect any type of traffic in egress and ingress. If an NVA approach is used, they can be found and deployed from Azure Marketplace.

We recommend that you use one set of Azure Firewall instances, or NVAs, for traffic originating on the internet. Use another for traffic originating on-premises. Using only one set of firewalls for both is a security risk as it provides no security perimeter

between the two sets of network traffic. Using separate firewall layers reduces the complexity of checking security rules, which makes it clear which rules correspond to which incoming network request.

[Azure Load Balancer](#) offers a high availability Layer 4 (TCP/UDP) service, which can distribute incoming traffic among service instances defined in a load-balanced set. Traffic sent to the load balancer from front-end endpoints (public IP endpoints or private IP endpoints) can be redistributed with or without address translation to a set of back-end IP address pools (such as network virtual appliances or virtual machines).

Azure Load Balancer can probe the health of various server instances. When an instance fails to respond to a probe, the load balancer stops sending traffic to the unhealthy instance. In a virtual datacenter, an external load balancer is deployed to the hub and the spokes. In the hub, the load balancer is used to efficiently route traffic across firewall instances. In the spokes, the load balancers are used to manage application traffic.

[Azure Front Door](#) (AFD) is Microsoft's highly available and scalable web application acceleration platform, global HTTP load balancer, application protection, and content delivery network. Running in more than 100 locations at the edge of Microsoft's Global Network, AFD enables you to build, operate, and scale out your dynamic web application and static content. AFD provides your application with world-class end-user performance, unified regional/stamp maintenance automation, BCDR automation, unified client/user information, caching, and service insights.

The platform offers:

- Performance, reliability, and support service-level agreements (SLAs).
- Compliance certifications.
- Auditable security practices that are developed, operated, and natively supported by Azure.

Azure Front Door also provides a web application firewall (WAF), which protects web applications from common vulnerabilities and exposures.

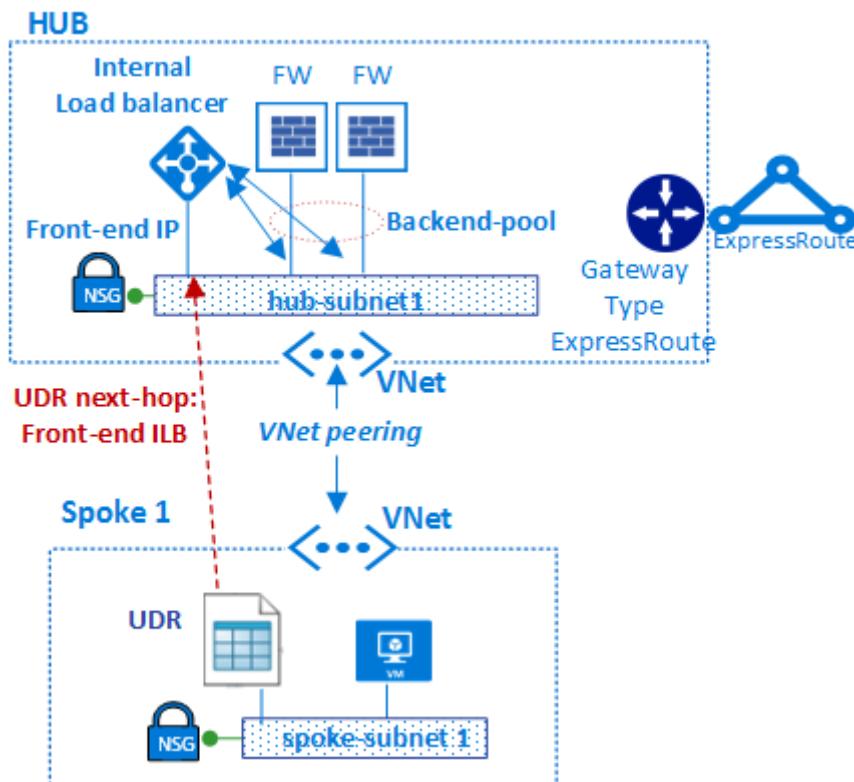
[Azure Application Gateway](#) is a dedicated virtual appliance providing a managed application delivery controller. It offers various Layer 7 load-balancing capabilities for your application. It allows you to optimize web farm performance by offloading CPU-intensive SSL termination to the application gateway. It also provides other Layer 7 routing capabilities, such as round-robin distribution of incoming traffic, cookie-based session affinity, URL-path-based routing, and the ability to host multiple websites behind a single application gateway. A web application firewall (WAF) is also provided as part of the application gateway WAF SKU. This SKU provides protection to web

applications from common web vulnerabilities and exploits. Application gateway can be configured as internet-facing gateway, internal-only gateway, or a combination of both.

Public IPs. With some Azure features, you can associate service endpoints to a public IP address so that your resource is accessible from the internet. This endpoint uses NAT to route traffic to the internal address and port on the virtual network in Azure. This path is the primary way for external traffic to pass into the virtual network. You can configure public IP addresses to determine which traffic is passed in and how and where it's translated onto the virtual network.

Azure DDoS Protection provides more mitigation capabilities over the **basic service tier** that are tuned specifically to Azure virtual network resources. DDoS Protection is simple to enable and requires no application changes. Protection policies are tuned through dedicated traffic monitoring and machine learning algorithms. Policies are applied to public IP addresses associated to resources deployed in virtual networks. Examples include Azure load balancer, Azure application gateway, and Azure service fabric instances. Near real-time, system-generated logs are available through Azure monitor views during an attack and for history. Application layer protection can be added through the Azure application gateway web application firewall. Protection is provided for IPv4 and IPv6 Azure public IP addresses.

The hub and spoke topology uses virtual network peering and user-defined routes to route traffic properly.



In the diagram, the user-defined route ensures that traffic flows from the spoke to the firewall before passing to on-premises through the ExpressRoute gateway (if the firewall

policy allows that flow).

Component type: monitoring

[Monitoring components](#) provide visibility and alerting from all the other component types. All teams can have access to monitoring for the components and services they have access to. If you have a centralized help desk or operations teams, they require integrated access to the data provided by these components.

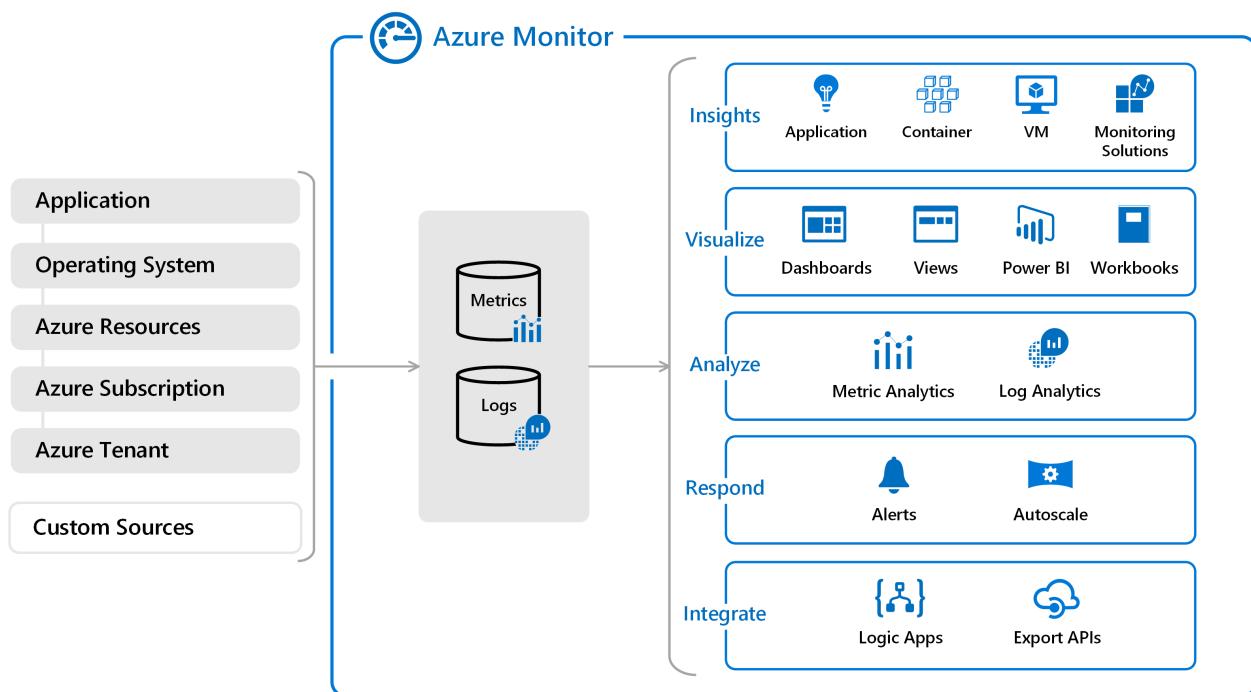
Azure offers different types of logging and monitoring services to track the behavior of Azure-hosted resources. Governance and control of workloads in Azure is based not just on collecting log data, but also on the ability to trigger actions based on specific reported events.

[Azure Monitor](#). Azure includes multiple services that individually perform a specific role or task in the monitoring space. Together, these services deliver a comprehensive solution for collecting, analyzing, and acting on system-generated logs from your applications and the Azure resources that support them. They can also work to monitor critical on-premises resources to provide a hybrid monitoring environment.

Understanding the tools and data that are available is the first step in developing a complete monitoring strategy for your applications.

There are two fundamental types of logs in Azure Monitor:

- [Metrics](#) are numerical values that describe some aspect of a system at a particular point in time. They're lightweight and capable of supporting near real-time scenarios. For many Azure resources, you'll see data collected by Azure Monitor right in their overview page in the Azure portal. As an example, look at any virtual machine and you'll see several charts displaying performance metrics. Select any of the graphs to open the data in metrics explorer in the Azure portal, which allows you to chart the values of multiple metrics over time. You can view the charts interactively or pin them to a dashboard to view them with other visualizations.
- [Logs](#) contain different kinds of data organized into records with different sets of properties for each type. Events and traces are stored as logs along with performance data, which can all be combined for analysis. Log data collected by Azure Monitor can be analyzed with queries to quickly retrieve, consolidate, and analyze collected data. Logs are stored and queried from [log analytics](#). You can create and test queries using log analytics in the Azure portal, and directly analyze the data using these tools or save queries for use with visualizations or alert rules.



Azure Monitor can collect data from various sources. You can think of monitoring data for your applications in tiers ranging from your application, any operating system, and the services it relies on, down to the Azure platform itself. Azure Monitor collects data from each of the following tiers:

- **Application monitoring data:** Data about the performance and functionality of the code you've written, regardless of its platform.
- **Guest OS monitoring data:** Data about the operating system on which your application is running. This OS could be running in Azure, another cloud, or on-premises.
- **Azure resource monitoring data:** Data about the operation of an Azure resource.
- **Azure subscription monitoring data:** Data about the operation and management of an Azure subscription, and the health and operation of Azure itself.
- **Azure tenant monitoring data:** Data about the operation of tenant-level Azure services, such as Microsoft Entra ID.
- **Custom sources:** Logs sent from on-premises sources can be included as well. Examples include on-premises server events or network device syslog output.

Monitoring data is only useful if it can increase your visibility into the operation of your computing environment. Azure Monitor includes several features and tools that provide valuable insights into your applications and other resources they depend on. Monitoring solutions and features such as application insights and Azure Monitor for containers provide deep insights into different aspects of your application and specific Azure services.

Monitoring solutions in Azure Monitor are packaged sets of logic that provide insights for a particular application or service. They include logic for collecting monitoring data

for the application or service, queries to analyze that data, and views for visualization. Monitoring solutions are available from Microsoft and partners to provide monitoring for various Azure services and other applications.

With such a collection of rich data, it's important to take proactive action on events happening in your environment, especially where manual queries alone won't suffice. Alerts in Azure Monitor proactively notify you of critical conditions and potentially attempt to take corrective action. Alert rules based on metrics provide near real-time alerting based on numeric values. Alert rules based on logs allow for complex logic across data from multiple sources. Alert rules in Azure Monitor use action groups, which contain unique sets of recipients and actions that can be shared across multiple rules. Based on your requirements, action groups can use webhooks that cause alerts to start external actions or integrate with your ITSM tools.

Azure Monitor also allows the creation of custom dashboards. Azure dashboards allow you to combine different kinds of data, including both metrics and logs, into a single pane in the Azure portal. You can optionally share the dashboard with other Azure users. Elements throughout Azure Monitor can be added to an Azure dashboard in addition to the output of any log query or metrics chart. For example, you can create a dashboard that combines tiles that show a graph of metrics, a table of activity logs, a usage chart from application insights, and the output of a log query.

Finally, Azure Monitor data is a native source for Power BI. Power BI is a business analytics service that provides interactive visualizations across various data sources. It's also an effective means of making data available to others within and outside your organization. You can configure Power BI to automatically import log data from Azure Monitor to take advantage of these more visualizations.

[Azure Network Watcher](#) provides tools to monitor, diagnose, and view metrics and enable or disable logs for resources in a virtual network in Azure. It's a multifaceted service that allows the following functionalities and more:

- Monitor communication between a virtual machine and an endpoint.
- View resources in a virtual network and their relationships.
- Diagnose network traffic filtering problems to or from a VM.
- Diagnose network routing problems from a VM.
- Diagnose outbound connections from a VM.
- Capture packets to and from a VM.
- Diagnose problems with a virtual network gateway and connections.
- Determine relative latencies between Azure regions and internet service providers.
- View security rules for a network interface.
- View network metrics.

- Analyze traffic to or from a network security group.
- View diagnostic logs for network resources.

Component type: Workloads

Workload components are where your actual applications and services reside. It's where your application development teams spend most of their time.

The workload possibilities are endless. The following are just a few of the possible workload types:

Internal applications: Line-of-business applications are critical to enterprise operations. These applications have some common characteristics:

- **Interactive:** Data is entered, and results or reports are returned.
- **Data-driven:** Data intensive with frequent access to databases or other storage.
- **Integrated:** Offer integration with other systems within or outside the organization.

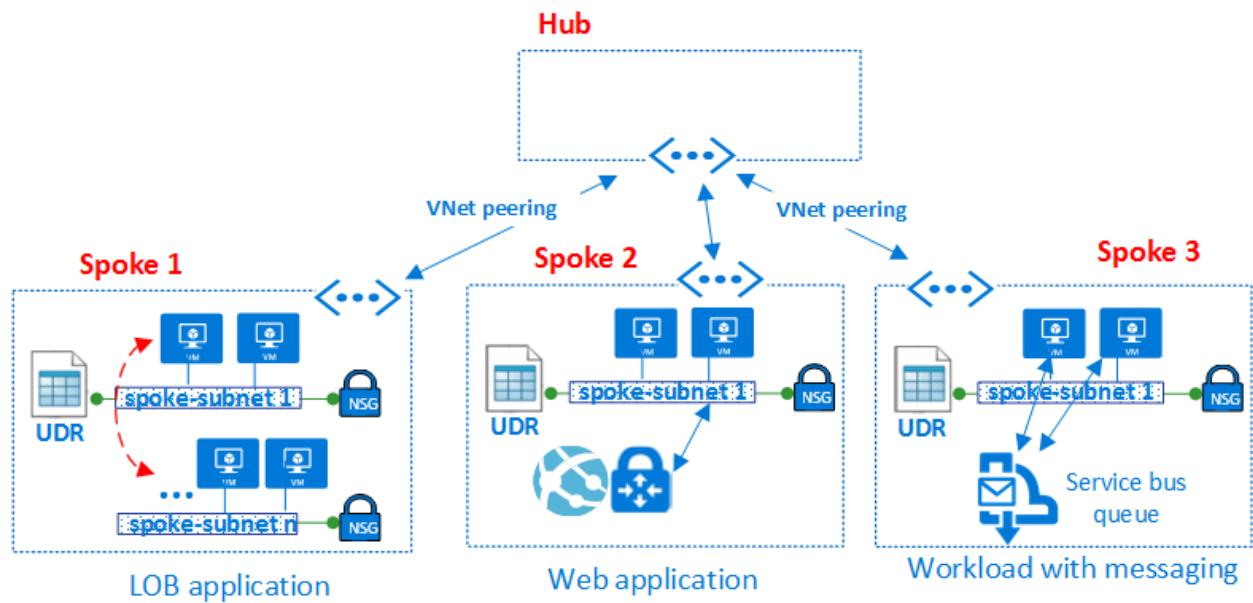
Customer-facing web sites (internet-facing or internally facing): Most internet applications are web sites. Azure can run a web site via either an IaaS virtual machine or an [Azure Web Apps](#) site (PaaS). Azure web apps integrate with virtual networks to deploy web apps in a spoke network zone. Internally facing web sites don't need to expose a public internet endpoint because the resources are accessible via private non-internet routable addresses from the private virtual network.

Big data analytics: When data needs to scale up to larger volumes, relational databases might not perform well under the extreme load or unstructured nature of the data.

[Azure HDInsight](#) is a managed, full-spectrum, open-source analytics service in the cloud for enterprises. You can use open-source frameworks such as Hadoop, Apache Spark, Apache Hive, LLAP, Apache Kafka, Apache Storm, and R. HDInsight. This supports deploying into a location-based virtual network, which can be deployed to a cluster in a spoke of the virtual datacenter.

Events and messaging: [Azure Event Hubs](#) is a big data streaming platform and event ingestion service. It can receive and process millions of events per second. It provides low latency and configurable time retention, enabling you to ingest massive amounts of data into Azure and read it from multiple applications. A single stream can support both real-time and batch-based pipelines.

You can implement a highly reliable cloud messaging service between applications and services through [Azure Service Bus](#). It offers asynchronous brokered messaging between client and server, structured first-in-first-out (FIFO) messaging, and publishes and subscribe capabilities.



These examples barely scratch the surface of the types of workloads you can create in Azure. You can create everything from a basic Web and SQL app to the latest in IoT, big data, machine learning, AI, and so much more.

Highly availability: multiple virtual datacenters

So far, this article has focused on the design of a single VDC, describing the basic components and architectures that contribute to resiliency. Azure features such as Azure Load Balancer, NVAs, availability zones, availability sets, scale sets, and other capabilities that help you include solid SLA levels into your production services.

However, because a virtual datacenter is typically implemented within a single region, it might be vulnerable to outages that affect the entire region. Customers that require high availability must protect the services through deployments of the same project in two or more VDC implementations deployed to different regions.

In addition to SLA concerns, several common scenarios benefit from running multiple virtual datacenters:

- Regional or global presence of your end users or partners.
- Disaster recovery requirements.
- A mechanism to divert traffic between datacenters for load or performance.

Regional/global presence

Azure datacenters exist in many regions worldwide. When selecting multiple Azure datacenters, consider two related factors: geographical distances and latency. To

optimize user experience, evaluate the distance between each virtual datacenter and the distance from each virtual datacenter to the end users.

An Azure region that hosts your virtual datacenter must conform with regulatory requirements of any legal jurisdiction under which your organization operates.

Disaster recovery

The design of a disaster recovery plan depends on the types of workloads and the ability to synchronize state of those workloads between different VDC implementations.

Ideally, most customers desire a fast fail-over mechanism, and this requirement might need application data synchronization between deployments running in multiple VDC implementations. However, when designing disaster recovery plans, it's important to consider that most applications are sensitive to the latency that can be caused by this data synchronization.

Synchronization and heartbeat monitoring of applications in different VDC implementations requires them to communicate over the network. Multiple VDC implementations in different regions can be connected through:

- Hub-to-hub communication built into Azure Virtual WAN hubs across regions in the same Virtual WAN.
- Virtual network peering to connect hubs across regions.
- ExpressRoute private peering, when the hubs in each VDC implementation are connected to the same ExpressRoute circuit.
- Multiple ExpressRoute circuits connected via your corporate backbone, and your multiple VDC implementations connected to the ExpressRoute circuits.
- Site-to-Site VPN connections between the hub zone of your VDC implementations in each Azure region.

Typically, Virtual WAN hubs, virtual network peering, or ExpressRoute connections are preferred for network connectivity, due to the higher bandwidth and consistent latency levels when passing through the Microsoft backbone.

Run network qualification tests to verify the latency and bandwidth of these connections, and decide whether synchronous or asynchronous data replication is appropriate based on the result. It's also important to weigh these results in view of the optimal recovery time objective (RTO).

Disaster recovery: diverting traffic from one region to another

Both [Azure Traffic Manager](#) and [Azure Front Door](#) periodically check the service health of listening endpoints in different VDC implementations. If those endpoints fail, Azure Traffic Manager and Azure Front Door route automatically to the next closest VDC. Traffic Manager uses real-time user measurements and DNS to route users to the closest (or next closest during failure). Azure Front Door is a reverse proxy at over 100 Microsoft backbone edge sites, using anycast to route users to the closest listening endpoint.

Summary

The virtual datacenter approach to migration is to create a scalable architecture that optimizes Azure resource use, lowers costs, and simplifies system governance. The virtual datacenter is typical based on hub and spoke network topologies (using either virtual network peering or Virtual WAN hubs). Common shared services provided in the hub, and specific applications and workloads are deployed in the spokes. The virtual datacenter also matches the structure of company roles, where different departments such as central IT, DevOps, and operations and maintenance all work together while performing their specific roles. The virtual datacenter supports migrating existing on-premises workloads to Azure, but also provides many advantages to cloud-native deployments.

References

Learn more about the Azure capabilities discussed in this document.

Network features

- [Azure Virtual Networks](#)
- [Network Security Groups](#)
- [Service Endpoints](#)
- [Private Link](#)
- [User-Defined Routes](#)
- [Network Virtual Appliances](#)
- [Public IP Addresses](#)
- [Azure DNS](#)

Load balancing

- [Azure Front Door](#)
- [Azure Load Balancer \(Layer 4\)](#)
- [Application Gateway \(Layer 7\)](#)
- [Azure Traffic Manager](#)

Connectivity

[Virtual Network Peering](#)
[Virtual Private Network](#)
[Virtual WAN](#)
[ExpressRoute](#)
[ExpressRoute Direct](#)

Identity

[Microsoft Entra ID](#)
[Microsoft Entra multifactor authentication](#)
[Azure role-based access control](#)
[Azure built-in roles](#)

Monitoring

[Network Watcher](#)
[Azure Monitor](#)
[Log Analytics](#)

Best practices

[Management Group](#)
[Subscription Management](#)
[Resource Group Management](#)
[Azure Subscription Limits](#)

Security

[Azure Firewall](#)
[Firewall Manager](#)
[Application Gateway WAF](#)
[Front Door WAF](#)
[Azure DDoS](#)

Other Azure services

[Azure Storage](#)
[Azure SQL](#)
[Azure Web Apps](#)
[Azure Cosmos DB](#)
[HDInsight](#)
[Event Hubs](#)
[Service Bus](#)
[Azure IoT](#)
[Azure Machine Learning](#)

Next steps

- Learn more about [virtual network peering](#), the core technology of hub and spoke topologies.
 - Implement [Microsoft Entra ID](#) to use [Azure role-based access control](#).
 - Develop a subscription and resource management model using Azure role-based access control that fits the structure, requirements, and policies of your organization. The most important activity is planning. Analyze how reorganizations, mergers, new product lines, and other considerations will affect your initial models to ensure you can scale to meet future needs and growth.
-

Feedback

Was this page helpful?

 Yes

 No